SYMBOLIC AND DISTRIBUTED WORD REPRESENTATIONS

MÁRTON MAKRAI

April 2022 – version 1

to my godfather Karcsi,
who never stopped urging me
to write this thesis

These results display two properties, one of them remarkable.

— *Levelt, Roelofs, and Meyer 1999*

ABSTRACT

This thesis connects distributed word models, more concretely word embeddings of shallow neural networks, to symbolic representations of lexical relations, verb argument roles, and cross-lingual word sense induction (the task of clustering word occurrences to lexical units). Our theoretical framework is the `4lang` semantic network.

CONTENTS

ACRONYMS

ACL The Association for Computational Linguistics

ACT Actions in CD, see Section 2.1.3

AGT Agent, verbal role, see Chapter 6

AI Artificial Intelligence, technically synonymous to ML. For its history, see Table 1

ALS Alternating Least Squares algorithm, see Section 7.3

AMR Abstract Meaning Representation, see Section 2.4.7

AT The locative case of static location, see Chapter 6

BERT A deep LM architecture, the most famous one, see Section 4.3

BPE Bite-pair encoding, mentioned in Sections 4.2.11.1 and 4.3.2

CAUSE A binary predicate used both by Jackendoff (Section 2.2.5), and in `4lang` (Section 5.5)

CCG Combinatorial Categorical Grammar, mentioned in Sections 2.4.7 and 2.4.9

CD Conceptual Dependencies, see Section 2.1.3

CED The Collins-COBUILD dictionary (Sinclair 1987), mentioned in Section 8.3

CG Conceptual Graphs, see Section 2.1.5

CPD Canonical Polyadic Decomposition, see Section 7.3

CS Conceptual Structures, see Section 2.2.5

DAG Directed Acyclic Graph, mentioned in Section 5.6

DAT Dative, verbal role, see Chapter 6

DO Direct Object

DST Dialogue State Tracking, see Section 4.2.10

EFNILEX A computational lexicographic project of the European Federation of National Institutions for Language

FCA Formal Concept Analysis, see Section 5.6

FE Frame Element, see Section 2.4.2

FOR One of the argument cases used for relational nouns in `4lang`, see Chapter 6

FROM The locative case of Source, see Chapter 6

GL The Generative Lexicon, see Section 2.2.7

GLUE A multitask benchmark for English, see Chapter 1

GMM Gaussian mixture models, see Section 7.4

GPT Graphical processing unit, a kind of harware used for deep learning

GS Grefenstette and Sadrzadeh (2011), see Section 7.4.3

HAS The binary relation of possession in `4lang`

HDBScan A hierarchical density-based clustering algorithom (McInnes, Healy, and Astels 2017)

HLBL Static word embeddings by (Mnih and G. E. Hinton 2009). We use them in Sections 5.4 and 5.5

HNC The Hungarian National Corpus (Oravecz, Váradi, and Sass 2014). We use it in Sections 5.3, 8.1 and 8.4

HPSG Head-Driven Phrase Structure Grammar, mentioned in Sections 2.2.7 and 2.4.9

HS Hierarchical softmax. We experiment with it in Section 5.3

KB Knowledge Base, see Section 2.3.3

KR Knowledge Representation, mentioned in Chapter 3

KS Kartsaklis and Sadrzadeh (2014), a benchmark dataset we use in Section 7.5

LDA Latent Dirichlet Allocation, see Section 4.1.3

LDOCE The Longman Dictionary of Current English (Section 2.3.4)

LDV Longman Defining Vocabulary, see Section 3.2

LFG Lexical Functional Grammar, mentioned in Sections 2.2.5 and 2.2.7

LM Language Model, see Section 4.2

LREC Intl Conference on Language Resources and Evaluation

LSA Latent Semantic Analysis, see Section 4.1.3

LSTM Long short-term memory, one of the major neural network architectures (Hochreiter and Schmidhuber 1997)

ML Machine learning

MLM Masked language modeling, see Section 4.3.2

MLP Multi-layer perceptron

MRD Machine Readable Dictionary, see Section 2.3.4

MSE Multi-sense embedding, see Section 8.2

MSZNY *Magyar Számítógépes Nyelvészeti Konferencia*, the Hungarian NLP conference

MT Machine Translation

NGD Normalized Google Distance, see Section 4.1.5

NLP Natural Language Processing

NMT Neural Machine Translation, see Section 4.3.4

NN Nearest Neighbor, not to be confused with Neural Networks

NP-hard A computational complexity class

NP Noun phrase (a concept in structuralist syntax)

NSM Natural Semantic Metalanguage, see Section 2.2.3

NSP Next sentence prediction, one of the pre-training tasks for BERT, see Section 4.3.2

OBL Oblique, verbal role, see Chapter 6

OSub Open Subtitles Corpus, see Sections 5.3.1.2 and 8.2

PAT Patient, verbal role, see Chapter 6

PCA Principlal Component Analysis, see Section 4.1.1

PDT The Prague Dependency Treebank

POS Part of speech

POSS Possessive, one of the argument cases used for relational nouns in `4lang`, see Chapter 6

(P)PMI (Positive) Pointwise Mutual Information, see Sections 4.1.2 and 7.2

PP Prepositional phrase

PTM Pre-trained model

REL The contentless argument relation in `4lang`, see Chapter 6

RNN Recurrent Neural Network,
one of the major neural network architectures

rNN Reverse Nearest Neighbor, see Section 8.2

RNNS2S RNN sequence-to-sequence model

SAT Boolean satisfiability problem

SEL Sense enumeration lexicons, see Section 2.2.7

SENNA Static word embeddings by (Collobert et al. 2011).
We use them in Sections 5.4 and 5.5

SGNS Skip-gram with negative sampling, see Section 4.2.5

SIF Smooth IDF (where IDF is Inverse Document Frequency).
We mention it in Section 4.3.6

SOTA State-of-the-art

SRL Semantic Role Labelling, see Chapters 2 and 6

SRT Semantic representats are abbreviated this way in the paper
discussed in Section 2.4.9

SVD Singular Value Decomposition, see Section 4.1.3

SVO Subject, verb, and object (Section 7.5.2),
especially in this order (Section 4.3.2)

TLC The Teachable Language Comprehender (Quillian 1969)

TO The locative case of Goal, see Chapter 6

UCCA Universal Conceptual Cognitive Annotation,
see Sections 2.4.7 and 2.4.9

UD Universal Dependencies, see Section 2.4.8

UMAP A manifold approximation method for dimension reduction
(McInnes et al. 2018)

uSIF A variant of SIF. We mention it in Section 4.3.6

VP Verb phrase

VSM Vector space models, introduced in Chapter 4

WMT The main conference on machine translation

WSD Word Sense Disambiguation, see Section 8.2

WSI Word Sense Induction, see Section 8.2

XLNET A deep language model

Innumerable roads lead to "knowledge,"
and we try to explore many of them.

— Findler (1979)

1 INTRODUCTION

Natural language processing is the engineering field of understanding texts (sentences, documents, etc.) produced by people, and generating texts for human reading. Computational linguistics, on the other hand, creates models of language with the motivation that the more a model works, the more it reflects the linguistic system. Computational representations of word meaning can be categorized as symbolic (especially semantic networks and logical formulas) or distributional (neural networks). The aims of alternative approaches are also diverse, ranging from compositionality and the syntax-semantics interface through logical aspects of meaning, to the relation between linguistic meaning and conceptual phenomena. The present thesis offers computational linguistics research submitted to a theoretical linguistics programme, while the author has a mathematical way of thinking mixed with psycholinguistic motivations.

The thesis is organized in two parts. The first three chapters give the *background* in symbolic representations (Chapter 2 in general, and Chapter 3 to the semantic network `4lang` of the research group the author belongs to) and distributional ones (Chapter 4). The second part describes the main contributions related to (possibly interlingual) *lexical relations* – relations that hold between the meanings of words independent of context. Chapter 5 starts with word analogies, translation, antonymy (opposite meaning), causality, and hypernymy (what basic category a word belongs to, e.g. *dogs* are *animals*)[1]. Chapter 6 and Chapter 7 investigate the semantic (a.k.a. thematic) *roles of verb arguments* in symbolic and distributional perspectives, respectively. Chapter 8 concludes the thesis with an evaluation proposal in the distributional study of *word ambiguity*. The table of contents at the beginning of the dissertation goes down to sections. There is also a mini table of contents at the beginning of each chapter, which go one step deeper, to subsections.

1 Sections in this chapter have appeared in proceedings of conferences, and here they appear in that same chronological order.

Part I

BACKGROUND

The first three chapters of the thesis give the *background* in symbolic representations (Chapter 2 in general, and Chapter 3 to the semantic network `4lang` of the research group the author belongs to) and distributional ones (Chapter 4).

Definition and word meaning need not have anything to do with grammaticalization or grammatical behavior. This is a fairly uninteresting claim about the relation between language and thought.

— *Pustejovsky (1995)*

2

SYMBOLIC REPRESENTATIONS

Contents

This chapter gives the background in semantic networks, the linguistic content that has to be represented, and computational resources for lexical semantics.

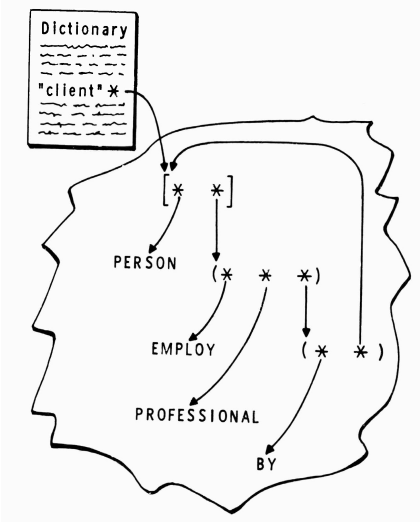

Figure 1: Associative links (Quillian 1968)

2.1 EARLY SEMANTIC NETWORKS

One of the main contributions of this thesis (Chapter 3) proposes a set of verb argument roles in an early version of the `4lang` semantic network. Sections 2.1.1 and 2.1.2 give the basics of semantic networks, while Section 2.1.6 introduces considerations about the so called definition graph. The three sections in between are more closely related to early Artificial Intelligence than to linguistics, and they have had a strong inpact on `4lang`.

2.1.1 *The Teachable Language Comprehender*

Quillian proposed a spreading-activation theory of human semantic processing, and tried to implement it in computer simulations of memory search, comprehension, and priming. In the description of the memory of the seminal Teachable Language Comprehender (TLC, Quillian (1969)) define text comprehension as relating assertions made or implied in some text to information previously stored as part of the comprehender's general knowledge of the world. Assertions in the text and permanent word knowledge are represented in TLC by the same format. TLC aims to understand general English texts without specific (mathematical or visual) reasoning rather than working in a restricted universe like SHRDLU (Winograd 1972). Here we describe the representation format of assertions, but not e.g. the syntactic component (consisting of so called form test) and the teaching protocol.

Figure 1 shows the memory unit representing *client*. Information is encoded as either a *unit* or as a *property*. Units (square brackets) represent objects and events while properties (parentheses) encode predications. Both brackets and parentheses are ordered lists of pointers

(asterisks) to other units or properties. The first pointer in a unit leads to some other unit referred to as that unit's *superset*. The remaining elements, if any, point to properties. Similarly to what we see in lexical decomposition (Section 2.2.1), the superset and these properties are analogous to the Aristotelian genus versus differentia specifica with the development that memory units in TLC represent not only lexical items but specific entities as what is asserted about them at some point of text comprehension:

> [A] concept is always represented in our format by pointing to some generic unit, its superset, of which it can be considered a special instance, and then pointing to properties stating how that superset must be modified in order to constitute the concept intended.

Properties are attribute-values pairs including traditional dimensions such as $(color, white)$ and dependency pairs (a theory by Tesniére (1959), formalized by Hays (1964)) such as $(on, hill)$ or $(employed, professional)$. The first element points to the attribute and the second to the value. These two obligatory elements are followed optionally by any number of pointers to other properties. The semantic content of attribute-value pairs is exemplified using *young client* where correct comprehension "must supply the fact that this client's 'age' is being judged young, which is not explicit in the text".

The network is responsible for inheritance between concepts, the computation of semantic relatedness, disambiguation, and anaphora resolution with a mechanism that gave rise to the whole theory of *spreading activation* in computational linguistics, to which we turn now.

2.1.2 *Spreading activation*

Simply put, spreading activation is a heuristic variant of shortest path finding and breadth-fist search in edge-weighted semantic networks with the psychological motivation of modeling semantic memory search and priming. Spreading activation experiments in the `4lang` framework have been published in Nemeskey et al. (2013). This subsection describes Collins and Loftus (1975)'s elaboration of Qullian's theory (shedding light on several misconceptions and offering additional assumptions). Collins and Loftus wanted to give an account of psycholinguistic experiments of their time. In their interpretation, "the full meaning of any concept is the whole network as entered from the concept node", quite reminiscent of the structuralist view on word meaning.

2.1.2.1 *An extension of the semantic memory to deal with psychological experiments*

Collins and Loftus extend Quillian's theory of semantic memory search and semantic priming in order to deal with a number of psychological

experiments. The resulting theory can also be considered as a prescription for building human semantic processing in a computer. They argue that the adequacy of a psychological theory should no longer be measured solely by its ability to predict experimental data: a theory should *produce* the behavior that it purports to explain.

QUILLIAN'S THEORY OF SEMANTIC MEMORY In their first section, Collins and Loftus try to correct a number of the common misunderstandings of the original theory. While the theory was developed as a program for a digital computer, Collins and Loftus elaborate it in psychological terms. People's concepts contain indefinitely large amounts of information, less and less relevant in a specific situation. Concepts (particular senses of words or phrases) can be represented as a node in a network, with properties of the concept represented as labeled relational links. Collins and Loftus specify a couple of properties of the links:

- Links usually go in both directions between two concepts.
- Links can have different *criterialities*, which are numbers indicating how essential each link is to the meaning of the concept. The criterialities in two directions can be different.
- The full meaning of any concept is the whole network as entered from the concept node.
- There are the following kinds of links:
  - superordinate (`isa`) and subordinate links,
  - modifier links,
  - disjunctive sets of links,[1]
  - conjunctive sets of links, and
  - a residual class of links, which allowed the specification of any relationship where the relationship (usually a verb relationship) itself was a concept.
- Links could be nested or embedded to any degree of depth.

Priming affects links as well as nodes.

Spreading activation means that search in memory between concepts involves tracing out in parallel (simulated in the computer by a breadth-first search) along the links from the node of each concept specified by the input words. The words might be part of a sentence or stimuli in an experimental task. At each node reached in this process, an activation tag is left that specifies the starting node and the immediate predecessor. If intersection between the two nodes has been found, by

1 `4lang` has no disjunction of edges.

following the tags back to both starting nodes, the *path* that led to the intersection can be reconstructed. The path is finally evaluated to decide if it satisfies the constraints imposed by syntax and context.

Collins and Loftus discuss common misinterpretations concerning Quillian's theory. The goal of the author of this thesis is not to decide these questions, just to show what specific problems arise if one wants do apply spreading activation. The questions may be answered on empirical grounds. There is no difficulty for Quillian's theory to adapt to either solution of the problems below.

- These is a stronger and a weaker version of the cognitive economy principle: "all properties are stored only once in memory and must be retrieved through a series of inferences for all words except those that they most directly define", vs "every time one learns that X is a bird, one does not at that time store all the properties of birds with X in memory".
- "All links are equal." In Quillian's original theory, there were criteriality tags on links, as we described earlier. Links were assumed to have differential accessibility (i.e. strength or travel time). The accessibility of a property depends on how often a person thinks about or uses a property of a concept. Whether criteriality and accessibility are treated as the same or different is a complex issue.
- Memory search (to make a categorization judgment) proceeds from the instance to the category. In a categorization task, response time is measured for a subject to decide whether or not a particular instance (e.g., *car*) is a member of one or more categories (e.g., *flower* or *vehicle*).
- "Search rate is slower in proportion to the number of paths that must be searched." vs "Independent parallel search is like a race where the speed of each runner is independent of the other runners which was a common assumption in psychology."
- Other misconceptions concern whether the network is a rigid hierarchy or whether the theory predicts it will always take less time to compare concepts that are close together in the semantic network.

THE EXTENDED THEORY In their next section, Collins and Loftus extend the theory with several assumptions to apply it to some psychological experiments (also transforming the theory from computer terms to quasi-neurological terms).

Local Processing Assumptions

- Activation spreads out along the paths of the network in a decreasing gradient. The decrease is inversely proportional to the accessibility or strength of the links.

- The longer a concept is continuously processed, the longer activation is released from the node of the concept at a fixed rate. In this model, only one concept can be actively processed at a time.
- Activation decreases over time and/or intervening activity.

  These assumptions impose a limitation on the amount of activation that can be allocated in priming more than one concept, because the more concepts that are primed, the less each will be primed.
- With the assumption that activation is a variable quantity, intersection requires a threshold for firing, and activation from different sources adds up.

Global Assumptions About Memory Structure and Processing are generalizations of earlier arguments that semantic memory is organized primarily into *noun categories* and that there is a *dictionary (or lexical memory)* separate from the conceptual network.

- The network is organized along the lines of semantic similarity. The more properties two concepts have in common, the more links there are between the two nodes. For example, different vehicles or different colors will all be highly interlinked, while red things (e.g., fire engines, cherries, sunsets, and roses) are not closely interlinked, despite the one property they have in common. Semantic relatedness is a slightly different notion from semantic distance, though the two terms are sometimes used interchangeably: distance is along the shortest path, and relatedness (or similarity) is an aggregate of all the paths.
- The names of concepts are stored in a lexical network (or dictionary) that is organized along lines of phonemic (and to some degree orthographic) similarity. People can identify these properties about words on the "tip of their tongue".
- People can control whether they prime the lexical network, the semantic network, or both.

Assumptions About Semantic Matching Process

A semantic matching process is the the categorization tasks, which ask "Is X a Y?". This process occurs in many aspects of language processing, such as matching referents, assigning cases, and answering questions.

- In order to decide whether or not a concept matches another concept, enough evidence must be collected to exceed either a positive or a negative criterion.
  - Evidence consists of various kinds of intersections that are found.

  - Evidence from different paths in memory sum together.
  - Positive and negative evidence act to cancel each other out.
  - Failure to reach either criterion before running out of relevant evidence leads to a 'don't know' response.

  This process is essentially the Bayesian decision model, which was common in the reaction time literature.

- If the memory search finds that there is a superordinate (or a negative superordinate) path from X to Y, that fact alone can push the decision over the positive (or negative) criterion. Superordinate links act like highly criterial property links.

- If the memory search finds properties on which X and Y (resp. mis)match (i.e. common properties), this is positive (resp. negative) evidence proportional, to the criteriality of the property for Y. Positive and negative evidence can be weighted differently: a mismatch on just one fairly criterial property can lead to a negative decision, whereas most of the highly criterial properties must match in order to reach a positive decision.

- Wittgenstein strategy is a variant of the property comparison strategy: to decide whether something is a game (for example, frisbee), a person compares it to similar instances that are known to be games. Here, matching properties count just as much toward a positive decision as distinguishing properties count toward a negative decision.

- Mutually exclusive subordinates strategy: if two concepts have a common superordinate with mutually exclusive links into the common superordinate, then this constitutes strong negative evidence, almost comparable to a negative superordinate link. Lacking specific information to the contrary, people may make a default assumption of mutual exclusivity when two concepts have a common superordinate.

- Counterexamples also can be used as negative evidence. E.g. "All birds are canaries" is disconfirmed by finding e.g. a robin. More formally: if the question is of the form "Is X a Y?" and there is a superordinate link from Y to X, and there exists Z that also has X as superordinate and is mutually exclusive from Y, this is conclusive evidence that X is not always a Y.

DEFINING AND CHARACTERISTIC FEATURES In the last section, Collins and Loftus deal with those aspects of semantic processing where the model of J. M. Smith (1974) is the major competitor to Quillian's theory. Smith represents concepts as bundles of semantic features of two kinds: defining and characteristic features. Defining features are those that an instance must have to be a member of the concept, and

features can be more or less defining. Characteristic features are those that are commonly associated with the concept, but are not necessary for concept membership. (The latter correspond to defaults in `4lang`.)

Categorization (decisions like "Is a car a flower?") consists of two stages. In Stage 1, all features are investigated, both characteristic and defining. If the match is above a positive criterion, the subject answers "yes"; if it is below a negative criterion, the subject answers "no"; and if it is in-between, the subject makes a second comparison, which is based on just the defining features. If the instance has all the defining features of the category, the subject says "yes".

The distinction between defining and characteristic features has the inherent difficulty, pointed out through the ages, that there is no feature that is absolutely necessary for any category.

> There is for living things a biologists' taxonomy, which categorizes objects using properties that are not always those most apparent to the layman. Thus, there are arbitrary, technical definitions that are different from the layman's ill-defined concepts, but this is not true in most domains. There is no technical definition of a game, a vehicle, or a country that is generally accepted.

Collins argues that

> the decision that a 'wren' is not a 'sparrow' would be made because they are mutually exclusive kinds of birds. They are both small songbirds, and it is hard to believe that many people know what the defining features of a sparrow are that a wren does not have. The fact that there are cases where people must use superordinate information to make correct categorization judgments makes it unlikely that they do not use such information in other cases.

If categorization consists of comparing features between the instance and the category, then it should not matter whether the instance or category is presented first, but experimental data shows asymmetry.

Another experiment that might show difficulties with the defining feature model is a categorization task of birds and animals on the one hand, and mammals and animals on the other. Deciding that bird names are in the category 'bird' is faster than that they are in the category 'animal,' whereas people are slower at deciding that mammal names are in the category 'mammal' than in the category 'animal'.

A final argument is that people have *incomplete knowledge* about the world: we often do not have stored particular superordinate links or criterial properties. Any realistic data base for a computer system will have this same kind of incomplete knowledge. The strongest criticism of the (J. M. Smith 1974) model is that it breaks down when people lack knowledge about defining features. By viewing superordinate links

as highly criterial properties, Quillian's extended theory encompasses a revised version of the Smith model as a special case of a more general procedure.

Levelt, Roelofs, and Meyer (1999) mention two other arguments against decomposition to features. The hypernym problem is that when a word's semantic features are active, then the feature sets for all of its hypernyms or superordinates are active. Still, there is no evidence that speakers tend to produce hypernyms of intended targets. The other argument is the lack of a semantic complexity effect: words with more complex feature sets are not harder to access (measured in reaction time).

2.1.3 *Let eleven verb-types bloom*

The most part of the preceding two sections have been about the formalism and the search heuristic implementing spreading activation. Now we turn to the semantic content of networks, first Conceptual Dependency (CD, Schank (1972)).

CD has been used by many computer programs of the time that understood English (MARGIE, the Script Applier Mechanism, and the Plan Applier Mechanism). From a linguistic point of view, CD is a meaning representation formalism which is inter-lingual, independent of paraphrase, and appropriate for drawing inferences.

In CD, the process of syntactic parsing is simultaneous with that of drawing some types of inferences. Schank (1973) distinguishes inference from logical deductions (i.e. those in automatic theorem proving). "The intent of inference-making is to 'fill out' a situation which is alluded by an utterance [and tie] pieces of information together to determine such things as feasibility, causality and intent of the utterance." While deductions are highly directed from axioms to some well-defined goal, inferences "are generally made 'to see what they can see' ". CD is a deep representation: the representation of a sentence including *buy a book* should include two actions of transfer, one whose object is the book and the other whose object is the price and the (roles of) participants in these actions. Default arguments (e.g. the object of the verb *drink* is alcoholic) are also subsumed, though the author notes that the presence of this default in many languages may be an artifact of shared culture, not that of the underlying (languages-independent) concept. Semantic arguments are meant broadly, e.g. the representation of *hit* should include the instrument. Assertions in CD graphs have a measure of confidence attached to them.

We describe the formalism of CD in some more detail as it has been very influential. There are conceptual categories:

- concepts of things that produce a picture (PP) of a real world item in the mind of the hearer, usually expressed by (common or proper) nouns,

- The birth of artificial intelligence (1952–1956)
- The golden years (1956–1974)
- The first AI winter (1974–1980)
- Boom (1980–1987): expert systems, knowledge, fifth generation computers, and connectionism
- Bust: the second AI winter (1987–1993)
- Application in industry and specific isolated problems (1993–2001)
- Deep learning, big data and artificial general intelligence: 2000–present

Table 1: Summers and winters of AI

- actions (ACTs) that are mostly expressed by verbs, and
- attributes modifying the former two (PA and AA, respectively).

The possible dependencies between concepts are specified by conceptual (relation) rules. Links may be modified for tense. To formulate dependency rules, verbs are "mapped into a conceptual construction that may use one or more [...] *primitive ACTs* in certain specified relationships plus other objects and states". Probably the most famous of these fourteen primitive ACTs are the three types of transfer, transfer of *abstract* relations e.g. ownership or control (ATRANS), that of *physical* objects (PTRANS), and that of information (*mental* transfer, MTRANS). These are related to the deep dative case `DAT` in `4lang`, see Section 6.2.2.2. In CD, there are four cases: OBJECTIVE, RECIPIENT, DIRECTIVE, and INSTRUMENTAL. We will return to these in Chapter 6.

Schank (1973) also discusses inferences that are independent of the specific language. Understanding the sentence *John told Mary that he wants a book* involves the inference that John wants the books for some MTRANS, and hearers of this sentence make the inference so spontaneously that they do not even remember whether this ACT was explicitly stated. An other example of the many types of inferences discussed are those about the reasons for actions (motivations of agents). The base for such inferences are so called *belief patterns*, sequences of causally-related ACTs and states that are shared by many speakers within a culture.

2.1.4 *What's in a link?*

The history of artificial intelligence (AI, including knowledge representation and connectionism) consists of summers and winters. A good summary is provided by the Contents of the Wikipedia page whose sections we cite (with a little modification) in Table 1.

Hubert Dreyfus argued that human intelligence and expertise depend primarily on unconscious instincts rather than conscious symbolic manipulation. Early approaches to artificial common-sense reasoning may seem so naive to the contemporary reader that the winter (which is

mostly measured in money) comes as no surprise in retroflection. The problems were made explicit by Woods (1975) dealing with the theoretical underpinnings of network representations and the semantics of the networks (nodes and links) themselves. He points out that despite the many publications and demonstrating systems, there is no theory of semantic networks, and existing networks are inadequate for the representations of many linguistic phenomena. Links have been used to represent what Brachman and Levesque (1985) call many different *levels* e.g. implementational pointers, logical relations, semantic relations (e.g., "cases"), and arbitrary conceptual and linguistic relations.

In section 2, Woods discusses what semantics is, whether it can be separated or even distinguished from syntax on the one hand and inference or "thought" on the other. In his terms, linguistics renders disambiguated representations to sentences while philosophy maps these to truth values. Retrieval and inference are not part of semantics, nor is pure disambiguation among syntactic parses, even if this is based on selectional restrictions and so-called semantic features. A system needs a separate semantic module for the justification calling it semantic.

The major characteristic of the semantic *networks* is the notion of links that may model human associations. Semantic representations need to be precise, formal, unambiguous, and logically adequate.[2] Woods discusses the problems of the existence of canonical form, the connection between attribute-value matrices and networks, relations of more than two arguments (a problem that is one of the main motivations of `4lang`, see the elimination of "deep ditransitives" in Section 3.1.3. Woods shows a prepositional example, x is 'between y and z'), and most importantly the logical type of nodes. Woods' Section 4 discusses two problems that are difficult for AI, restrictive relative clauses, intensional entities (representations of entities without commitment to existence or distinctness), and quantification. Most of these were also unsolved problems in the version of `4lang` on which the contributions of this thesis are based, but see Kornai (2022).

2.1.5 *Conceptual Graphs*

We turn to Conceptual Graphs (CG, J. Sowa (1976)), "two-dimensional form of logic", that connect semantic networks discussed so far to the broader discipline of knowledge representation and logic. An excellent introduction is offered by John F Sowa (1992).

CG is a knowledge representation language designed as a synthesis of semantic networks; "logic-based techniques of unification, lambda calculus, and Peirce's existential graphs; linguistic research based on

2 Logic is one of the main disciplines for meaning representation besides semantic networks and vector-space models. In this chapter, we assume familiarity with first order and intentional logic, and do not go into details as this is not necessary for the main chapters.

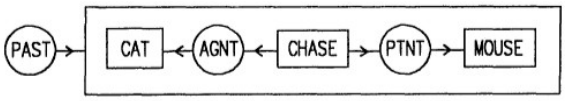

Figure 2: CS graph for *A cat chased a mouse* John F Sowa 1992, p.80

Tesniere's dependency graphs and various forms of case grammar and thematic relations; and data-flow diagrams and Petri nets, which provide a computational mechanism for relating conceptual graphs to external procedures and databases." The result is an expressive system of logic with a direct mapping between natural languages and e.g. expert systems. By combining Peirce's contexts with the dependency graphs, CG provides a formalism that can represent Schank's scripts.

As exemplified in Figure 2, CG represents concepts by rectangular nodes and dependency relations ("conceptual" relations) by circular ones as a typed (a.k.a. sorted) version of logic. (As we have already seen, Schank's graphs show conceptual relations as various kinds of arrows instead of these labeled circles.)

2.1.6 *The naive physics manifesto*

Hayes (1979) proposes the construction of a formalization of a portion of common-sense knowledge about the everyday physical world (objects, shape, space, movement, substances, time, etc.) along with a theory of meaning. The main characteristics of the proposed theory are

- thoroughness, i.e. coverage,
- fidelity: the theory should be reasonably detailed,
- density: the ratio of facts to concepts needs to be fairly high (i.e. the units have to have lots of slots), and
- uniformity: a common formal framework (language, system, etc.) so that the inferential connections between the different parts (axioms, frames,...) can be clearly seen. It is methodologically important to allow the use of a variety of formalisms in sub-areas, but idiosyncratic formalisms should be systematically reducible to the basic formalism, and be regarded as 'semantic sugar'.

In this section, we introduce sections 3 to 6 of Hayes (1979). For modern advances in this direction, see Hobbs (2008), which we discuss in Section 2.4.6.

THE AXIOM-CONCEPT GRAPH: CLUSTERS AND DENSITY A naive physics formalization consists of many assertions and symbols (i.e. tokens: relation symbols, function and constant symbols) – or: frame headers, slot names, etc.; or: node and arc labels, etc. The meaning of the tokens is defined by the structure of the formalization, by

the pattern of inferential connections between the assertions. The formalization is dense, if for each token, there are many axioms involving it, which pin down the meanings of the tokens. This view of meaning differs profoundly from the view which holds that tokens in a formalization are words in a natural language.

The axiom-concept (a-c) hypergraph consists of nodes corresponding to tokens of the formalization; and arcs corresponding to axioms: an arc links the tokens that it uses. The formalization is dense if the a-c graph is highly connected. Hayes does not expect density to be uniform: there will be more dense clusters of concepts. Identifying these clusters is one of the most important and difficult tasks. E.g. what happens with liquids, is part of the liquids cluster, not part of some theory of 'what-happens-when': causality is not a cluster. Cluster identification is hard, since a large conceptual structure can be entered anywhere. If it seems hard to say anything very useful about the concepts, that can mean that one has entered the graph at a locally sparse place, rather than in a cluster. This thesis analyses a similar graphical representation of `4lang` and its connected components in Section 3.3.

Clustering is hierarchical: e.g. the collection of concepts to do with three-dimensional shape and orientation ('above', 'below', 'tall', 'fat', 'wide', 'behind', 'touching', 'resting on', 'angle of slope', 'edge' (of a surface), 'surface' (of a volume), 'side', 'vertical', 'top', 'bottom', which have many internal relationships) must appear significantly in conceptual frameworks that underlie visual perception and locomotion, describing assemblies, the theory of liquids, and that of physical actions and events.

THE a/c RATIO AND REDUCTIONIST FORMALIZATIONS The ratio of axioms to concepts (the a/c ratio) will be large for a dense axiomatization. Any interesting axiomatization will have a/c greater than one; but there are interesting axiomatizations in which a/c will be very close to 1. E.g. in the Zermelo-Fraenkel set theory, $c = 2$ (the concepts are '$\in$' and 'set') and $a = 8$. This theory enables one to define many concepts (e.g. the integers; the rationals; the reals), and the desired properties of these concepts (e.g. the principle of induction for integers or the continuity of the real line) follow from the structure of these definitions, and the axioms as theorems of the axiomatization. The axiomatic approach to naive physics which Hayes proposes is different. Set theory is reductionist in the extreme it is extraordinarily sparse. By adding definitions to a reduced theory, a/c tends asymptotically to unity. The resulting a-c graph has one very small cluster at the center, surrounded by a cloud of nodes each linked radially. This reductionist graph is quite a different 'shape' from the connected, clustered graph of a dense axiomatic theory. Hayes believes that there is no such small, reductionist theory for common sense reasoning.

Many approaches in the artificial intelligence literature, make a reductionist assumption or 'semantic primitives', exemplified by the work of Wilks (1977) and Schank (1975). The number of primitives is about 90 in Wilks, and 14 in Schank. Schank and his students associated inference molecules with the 14 primitive action-tokens, which play the same sort of central organizing role that the set axioms do. The desired properties of e.g. buying or giving follow from their definitions, and the meaning given to the primitives by the core theory. Hayes criticizes Wilks for merely presenting a list of tokens with a brief description, i.e. the semantic primitives being English words. A reductionist, semantic-primitives based approach to meaning may be adequate for information-retrieval or machine translation, but at some point we will have to represent detailed knowledge of the world

MEANINGS, MODEL THEORY, AND FIDELITY If the meanings of tokens are not specified by definitions, then how? A token means a concept to the extent that the formalization enables a sufficient number of inferences to be made whose conclusions contain the token. But Hayes assumes that a formalization has an adequate model theory as well, i.e. tokens have extension. Hayes highlights the widespread delusion of confusing a formal description of a model found in the textbooks with the actual model. If axiomatization has a very much simpler model than the intended one, then the tokens mean no more than they mean in the simple model. This is what Hayes means by 'fidelity'. E.g. an adequate formalization of a blocks world will be such that any model of it must have an essentially three-dimensional structure. Fidelity is how closely the simplest model resembles the intended one.

A related problem is that the meaning of a token depends upon the entire formalization, a *change* to any part of the formalization can change every other part. People with different formalizations in their heads may understand the same token in different ways. Find a substance and a set of circumstances such that I would call it 'water' and you would not! It is even possible when our beliefs about water (i.e. all the assertions which actually contain the token 'water') are identical. The difference may lie in some related concept (such as viscosity, or drinkability) which we understand differently. It may not even be possible to say exactly which tokens we differ on. One of the good reasons for choosing naive physics to tackle first is that there seems to be a greater measure of interpersonal agreement here.

If you change the meaning of 'water', the change in the meanings of other tokens is less, the further away the token is from 'water'. As a working hypothesis, you may identify this distance with shortest-path distance in the axiom-concept hypergraph. Thanks to this distance-dilution effect, it seems a reasonable strategy to, first, work on clusters more or less independently. You can introduce concepts, which occur in some other cluster, fairly freely, assuming that their meaning is rea-

sonably tightly specified there. E.g. in considering liquids, I needed to talk about volumetric shape: our concept of a horizontal surface would hardly be complete if we had never seen a large, still body of water — but we assume of a fairly autonomous theory of shape. The 'definitions' view of meaning is theoretically wrong, but a good method. Finally, Hayes talks about the body and sensory input. As any consistent first-order axiomatization has a model with only symbols, 'motor tokens' — symbols which describe bodily movements — should directly be related to the body.

THOROUGHNESS AND CLOSURE One way to have a high a/c ratio, it might seem, would be to keep c small: find some small, self-contained groups of concepts which could be formalized in total isolation to a reasonable degree of fidelity. But in a typical situation, one quickly needs to introduce tokens, and in order to pin down their meanings, yet more concepts. The proliferation of tokens seems to be getting out of hand. If one thinks of exploring the a-c graph, one needs a sense of direction, to stay within the current cluster. During the formalization process, the proliferation must slow down eventually. The 'thoroughness' requirement is to go on until this slows down, when our collection of concepts has closed upon itself, so that all the things one wants to say in the formalization can be said using the tokens which have already been introduced. This means we have spanned the entire graph, and need only to add new arcs, filling out the graph until its density is sufficient to capture the meanings of its tokens. Hayes's program is to get a formalization which is closed and has high fidelity (so, high density): then it must, also be thorough.

To achieve greater fidelity, one will need greater thoroughness. E.g. to really capture the notion of 'above', you probably have to go into analogies to do with e.g. interpersonal status: (Judge's seats are raised; Heaven is high, Hell is low; to express submission, lower yourself, etc.) Imagine a world in which the 'status' analogy was reversed. That is a possible model of naive physics, but not of common sense. A formalization cannot be deep without being broad, and must be deep to be dense: so a dense formalization must be deep and broad. The cluster hierarchy mentioned before depends upon the fidelity, the level of detail. The programme of tackling naive physics in isolation is based on the belief that there is a level of detail at which naive physics forms a close cluster in a rich but tractable level of detail.

2.2 COGNITIVE SEMANTICS

Now we introduce Katz and Fodor (1963)'s seminal paper in lexical decomposition, and a line of semantic research that Kornai (2011) describes as "the less formally stated, but often strikingly insightful work in linguistic semantics" exemplified by the work of Wierzbicka (1985,

Section 2.2.3), Lakoff, Fauconnier, Langacker (1987), Talmy (1988, Section 2.2.4), Jackendoff (1990, Section 2.2.5), and others "often broadly grouped together as 'cognitively inspired' ". (References to sections in the present thesis added.) In Baroni and Lenci (2010)'s reflection, cognitive science and linguistics typically represent concepts as clusters of properties (Section 2.2.5): noun properties known as qualia roles (Section 2.2.7), verb selectional preferences and argument alternations (Section 2.2.6), event types, and "topical" relatedness between words, e.g. the relation between *dog* and *fidelity*.

2.2.1 *Lexical decomposition*

We start our account of computational lexical semantics with the standard model of lexical decomposition due to Katz and Fodor (1963). The paper describes its aim as the organization of facts contributed by diverse fields including philosophy, linguistics, philology, and psychology. The first part of the paper describes the domain, the descriptive and explanatory goals, the mechanisms, and the empirical and methodological constraints upon a semantic theory. They want to find a balance of strict formalism (developed some years later in Montague Grammar) and great explanatory power (like traditional lexicography). The input to their semantic model is a sentence analyzed by a recursive compositional grammar, in modern terms, a parse tree. These authors require a semantic theory be capable of recognizing (and resolving) ambiguity, paraphrase, and anomaly (e.g. *The paint is silent*) but other aspects like the computation of truth values is deferred. The difference between syntax and semantics is that the latter may rely on context, mainly linguistic one (the dialog), and to a restricted degree, extra-linguistic one (word knowledge). Their notion of word knowledge subsumes facts like 'buildings do not jump', which is needed for comprehending the sentences *Joe jumped higher than the Empire State Building* and *Joe jumped higher than you* differently. The theory should "interpret discourses just so far as the interpretation is determined by grammatical and semantic relations which obtain within and among the sentences of the discourse."

The components of the proposed semantic theory include the dictionary (the same module we will call the lexicon) and one that could be called a word-sense disambiguation method in present-day terms. The most important part of this theory is the structure of dictionary entries. Besides part of speech (POS) specification and, optionally, explicit cross-references to synonyms, dictionary entries consist of sense characterizations like that in Figure 3. The key notion is that of the semantic markers (in parenthesis, e.g. $(Human)$) that represent relations between meanings of the same polysemous word and between different dictionary entries. Distinguishers (in brackets) assigned to a lexical item are intended to reflect what is idiosyncratic about its meaning. This

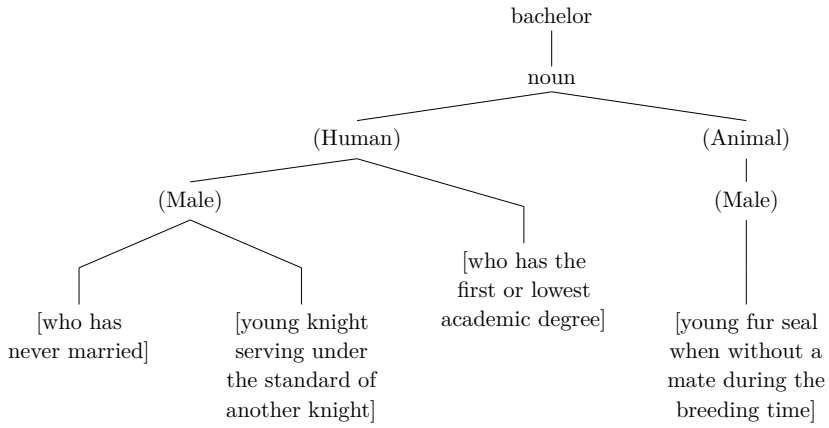

Figure 3: The sense-characterization of *bachelor* by Katz and Fodor (1963)

distinction is analogous (Kornai 2019, Chapter 5) to the Aristotelian notion of genus (a *mirror* is a 'plain surface') and a *differentia specifica* (. . . that 'reflects'). The unenclosed elements are grammatical "markers" (features). Semantics markers play a role in disambiguation, selectional restrictions, and, in a limiting case of selectional restrictions, the detection of semantic anomaly. The formalism also allows restrictions for the arguments of the items, e.g. $\langle(Female)\rangle$ in the representation of one of the senses of *honest* designates that the corresponding meaning of *honest* applies only to arguments with the $(Female)$ marker. In the concluding section, the authors mention that there may exist an universal inventory of semantic markers from which the markers of each particular language are drawn, a goal `4lang` shares with this theory (besides lexical decomposition itself).

2.2.2 *Case Grammar*

One of the main chapters of this thesis (Chapter 6) introduces the semantic roles used in `4lang`, the concept network of the research group the author belongs to. Our system has been heavily influenced by Case Grammar (Fillmore 1968), which this section introduces.

Our introduction is based on Palmer, Gildea, and Xue (2010, Chapter 1), who investigates semantic roles (semantic relations and predicate-argument structure) and the controversies surrounding them. They start with the example that from a sentence like *John threw a ball to Mary in the park*, an NLP system should identify a throwing event, John as the Agent or Causer of the event, Mary as the Recipient, the ball as the item being thrown, and the location of the throwing event. The linguistic theory of mapping from the syntactic analysis of the sentence to the underlying predicate argument structures is known as Linking. On the syntactic side, we have alternations like *John broke the window/The window broke* with the same semantic role (or concep-

tual relation) in both sentences. (In the example, it would typically be labeled as the Patient.)

Case Grammar originated with Fillmore's paper on "deep" cases, i.e. semantically typed verb arguments (Fillmore 1968). The theory involves types of nouns with different types of cases, e.g. the Agentive and Dative roles are most likely to be of type animate. Argument frames specify the number, type and obligatory/optional nature of roles associated with a verb. Linguists developed tests for determining whether two noun phrases have the same case. For instance, members of a conjunction have the same case. Representing alternative role assignments (e.g. *Mother is cooking the potatoes/The potatoes are cooking/Mother is cooking*) by the same deep cases can result in a more compact lexicon. Even *like* and *please* can be considered semantically equivalent, distinguished only by their preferred mappings. Within the semantic domain, generalizations can be exploited in the form of commonalities e.g. between the Agentive cases and the Objective cases of actions such as hitting, breaking, and cutting.

The inventory of roles differ between flavors of the theory, only the Agent and the Patient being relatively straightforward. The Agent is the initiator of the action, the doer, typically acting deliberately or on purpose. The question *What did X do?* can be applied, with X being the Agent. The Patient, on the other hand, is being acted upon. It is likely to change state as a result of the Agent's actions. The questions *What happened to Y?* or *What did X do to Y?* would apply.

2.2.3 *Natural Semantic Metalanguage*

Lenat and Guha (1990) formulate one of the greatest problems remaining in modern semantic networks as follows.

> Programs often use names for concepts such as predicates, variables, etc., that are meaningful to humans examining the code; however, only a shadow of that rich meaning is accessible to the program itself. For example, there might be some rules that conclude assertions of the form `laysEggsInWater(x)`, and other rules triggered off of that predicate, but that is only a fragment of what a human can read into `laysEggsInWater`

A solution to the problem of arbitrary node-labels has been offered outside of the computational realm, by the Natural Semantic Metalanguage (NSM) approach (Wierzbicka 1972) that we introduce following the first two chapters in a more recent collection (Goddard and Wierzbicka 1994), "the first attempt ever to empirically test a hypothetical set of semantic and lexical universals across a number of genetically and typologically diverse languages" with "parallel and strictly compa-

rable answers to the [questions of] a shared set of concepts, forming the common conceptual foundation of all cultures".

The principles of the work are specified in their Section 1.1 and enumerated below. Goddard includes a discussion of the opinion of the main semanticist of the century on these principles.

1. Semiotic Principle. "A sign cannot be reduced to or analyzed into any combination of things which are not themselves signs". Goddard lists some examples to what meaning can*not* be decomposed: reference or denotation, truth conditions, neurophysiological data, and usage. This principle is opposite to the goal of this thesis that searches for connections between symbolic representations and distributional ones.

2. Decomposition into discrete terms without (circularity and) residue. Exhaustive analysis distinguishes NSM from *componential analysis* which attempts to capture only *systematic oppositions* or "Katz and Jackendoff, who both believe that for many words an unanalysable residue of meaning remains" (the Distinguishers, recall Section 2.2.1). Commitment to discrete terms distinguishes NSM from *scalar notations*, the topic of Chapter 5.

3. Semantic Primitives Principle. There exists a finite set of undecomposable meanings and semantic primitives have an elementary syntax whereby they combine to form 'simple propositions'. While this is a key point of the collection, `4lang` does not require the elements of the core/defining vocabulary to be primitive.

4. NSM approach. "The proper metalanguage of semantic representation is [...] a minimal subset of ordinary natural language." Goddard lists position in the literature regarding the problem of the (meta-)semantics of the representational elements.

    - Proposals which represent primitives by *obscure technical terms* like symbols borrowed from logic ($\exists, \forall$) or those like Schank's PACT, CACT and TACT that need explanation in ordinary English (e.g. 'physical act', 'communication act' and 'transfer act', respectively).
    - Predicates in generative semantics, like CAUSE, NOT, BECOME and ALIVE whose intended meanings were not (exactly) those of the English words, but were more 'abstract'
    - Katz (1987) uses semi-technical labels to identify the 'conceptual components' of e.g. the English word CHASE: 'Activity', 'Physical', 'Movement', 'Fast', 'Direction', 'Toward location of', 'Purpose', 'Catching'. It has to be made clear what we gain by formalization as opposed to natural syntax.

5. "The NSMs derived from various languages will [...] have the same expressive power."

6. The linguistic exponents of semantically primitive meanings in different languages can be placed into one-to-one correspondence (modulo differences like) allolexy and POS membership, thus they share a common set of combinatorial properties.

7. Strong Lexicalization Hypothesis. Every semantically primitive meaning can be expressed through a word, morpheme or fixed phrase in every language. Exponents may be homonyms with different POSs or bound morphemes. Goddard follows Chomsky (1965) in distinguishing *formal universals* concerning the principles by which sense-components are combined to yield the meanings of lexemes from *substantive universals* concerning the identity of semantic components. The collection tests the thesis of the most extreme form of substantive universalism that "there is a fixed set of semantic components, which are lexicalised in all languages".

In the NSM approach, words (morphemes, etc.) can be identical in meaning despite different POS, ranges of use, or patterns of polysemy. differences in range of use does not invalidate the claim of semantic equivalence, as far as it is caused just by lexical blocking or social and cultural factors. the project has introduced *canonical contexts* to specify the sense of polysemous words that should be used in explications, and the contexts in which the proposed meanings is expected to be found. Canonical context, unlike explications, can be downloaded from the project site.

As admitted in the last chapter, the greatest problem with the NSM approach is polysemy as basic, everyday words are particularly likely to be polysemous because of Zipf's law. They require polysemy always to be justified on language-internal grounds, and to prove that a word is polysemous, one has to demonstrate that the putative senses call for distinct reductive paraphrase explications or syntactic frames (and distribution). Patterns of polysemy show similarities among languages (Youn et al. 2016).

Similarly to `4lang`, NSM has to reconcile the existence of language-specific morphosyntactic categories with the claim that the semantic metalanguage is isomorphic across languages, e.g. in the natural semantic metalanguage based on Latin, VOLO would never occur without an explicit subject. We will discuss this problem in Chapter 6.

Section 2.2 investigates The Proposed Primitive Inventory in the groups shown in Table 2. Albeit we are not interested in whether an element is primitive, it is useful to discuss how the core definitions in `4lang` handle the areas where these groups have proved indispensable in NSM.

Substantives	I, you, someone, something, people
Mental predicates	think, say, know, feel, want
Determiners/quantifiers	this, the same, other, one, two, many, all
Actions/events	do, happen
Meta-predicates	no, if, can, like, because, very
Time/place	when, where, after, before, under, above
Partonomy/taxonomy	have parts, kind of
Evaluators/descriptors	good, bad, big, small

Table 2: Primitives of Natural Semantic Metalanguage in groups.

2.2.4 *Force dynamics in language and cognition*

Talmy (1988) draws the attention to what he calls force dynamics: linguistic, psychological, and social phenomena related to physical ones, like the exertion of force, resistance to such exertion and the overcoming of such resistance, blockage of a force and the removal of such blockage, etc. Talmy offers a framework that also includes *letting*, *hindering*, *helping*. The theory builds upon the parallelisms between how we refer to physical and psychosocial matters.

In English, force dynamics is present in different grammatical categories: closed-class words (conjunctions, prepositions, modals), open-class lexical items, semantics of course (physical force psychological and social interactions, psychosocial "pressures"), and discourse (patterns of argumentation, discourse expectations and their reversal). The theory brings these together into systematic relationships.

Talmy attributes his method to "cognitive semantics" or "cognitive linguistics", which analyzes the cognitive process and its surface linguistic realizations together. Force dynamics is among the fundamental notional categories that languages use to structure and organize meaning, while they exclude other notional categories from this role. For cognitive semantics, it is important, how the linguistic structuring relates to perceptual modalities and reasoning, space, time, and visual perception, or, in this case, physics and psychology. The paper goes from conceptually basic physics dynamics to psychological and social interactions, the grammatical category of modals, discourse factors (argumentation), and other cognitive and conceptual domains.

The simplest force dynamics model consists of the following:

- two forces, and an Agonist and an Antagonist. The salient issue is whether the Agonist is able to manifest its force.
- The Agonist is toward action or toward inaction. The Antagonist is opposite that of the Agonist.

- The relative strengths of the Agonist and the Antagonist is a third parameter.
- The result is either action or inaction.

More complex force-dynamic patterns change through time: a stronger Antagonist can come in or go out, or the balance of forces can shift.

An additional kind of pattern is in which the Antagonist remains away. Corresponding to each of the steady-state patterns introduced so far, there is a secondary steady-state pattern with the Antagonist steadily disengaged. E.g. where the Antagonist is stronger, we have the patterns for the Antagonist *letting* the Agonist to move or rest.

There are alternatives of Foregrounding different subsets of the factors, e.g. making the Agonist, the Antagonist, or the result the subject or the object.

Examples with a weaker Antagonist: with the Agonist as the subject: *despite, although*, with the Antagonist as subject: *hinder, help, leave alone*.

*Psycho*dynamics generalizes notions of physical pushing, blocking to wanting and refraining; psychological 'pressure', and 'pushing'. The self may be divided to an Agonist and an Antagonist, where the Agonist represents the desires, and the Agonist is supression. In language, this is extended to physical entities without sentience such as wind, a dam, or a rolling log. A psychological component is normally included and understood as the factor that renders the stronger participant. The body has an intrinsic tendency toward rest, requiring animation by the psyche.

Two additional factors are the *phase* along a temporal sequence, and 'factivity': the occurrence or non-occurrence of portions of the sequence and the speaker's knowledge about this. With the Antagonist's as subject: *try* involves focus at the initial phase without knowledge of its outcome, while *succeed* and *fail* focus on a known occurrent or non-occurrent outcome.

The force dynamics in discourse (argumentation and expectations) is based on the metaphor of an *argument space*: each point can oppose or reinforce another point, and each encounter can move the argument state closer to or further from one of the opposing conclusions.

Another part of the paper compares conceptual models of physics implicit in language to the real physical theory. One great difference is the asymmetry between the privileged Agonist and the Antagonist so natural in language-based conceptualizing, which has no counterpart in physical theory. The real theory is based on objects' impetus in motion, while the naive theory assumes a tendency e.g. to come to rest. In modern physics, stationariness is not a distinct state but is simply zero velocity. In language either the Agonist or the Antagonist has greater relative strength, while in physics, two interacting objects must be exerting equal force. The linguistic expression of causation has a tripartite

structure: a static prior state, a discrete state-transition, and a static subsequent state. This is based on the notion of an 'event': a portion conceptually partitioned out of the continuum of occurrence, which is autonomous, without causal process during its occurrence. Blocking and letting, resistance and overcoming, some of the most basic force-dynamic concepts, have no principled counterpart in physics, because these concepts depend on the ascription of entityhood to a conceptually delimited portion of space, and the entity's intrinsic tendency toward motion or rest.

2.2.5 *Conceptual Structures*

"I think that an the overview of the ideas about the nature of argument structures and the mechanisms that lead from semantics argument structures to syntactic arguments", which will be investigated in Chapter 6, "must not miss Ray Jackendoff's proposal, which evolved in many articles and books since cca. the mid-1980s; if only because the notion of the lexical conceptual structure to be distinguished from the semantic structure is also relied on by those who otherwise propose different mapping mechanisms from that of Jackendoff (e.g. the Levin–Rappaport pair). Jackendoff (1990) is a relatively early (and, thankfully, fairly easy to understand) review of his views. You don't have to 'learn' this, but getting to know the basic ideas (it is enough to just go through the first 60 pages) can, in my opinion, get everyone to rethink new perspectives" (András Komlósy, personal communication, translated from Hungarian by thesis author).

"Building on ideas about semantics first expounded by Gruber (1965), Jackendoff (1972, 1983) elaborated significantly on the notion of cases by treating them as arguments to a set of *primitive conceptual predicates* such as `GO`, `BE`, `STAY`, `LET`, and `CAUSE`." (Palmer, Gildea, and Xue 2010)

`GO` can be used to describe changes of location, possession, or state, in any situation where both a "before state" and a different "after state" can be defined. It basically takes three arguments, the object undergoing the change and the before and after locations, possessors, or states. Later versions introduced subtypes of primitive predicates that add more information, e.g. the manner of a motion. Jackendoff 's intent was not to provide detailed representations of all of meaning but, to focus on the *mapping* between syntax and semantics. The remainder of this section discusses the theory based on Jackendoff (1990).

ONTOLOGICAL CATEGORIES OR CONCEPTUAL PARTS OF SPEECH

Instead of a division of formal entities into logical types like constants, variables, predicates, and quantifiers, the theory of Conceptual Structures (CS) sorts constituents to a few major ontological categories (or

conceptual parts of speech) like Thing, Event, State, Action, Place, Path, Property, and Amount.

Each major syntactic constituent maps into a conceptual constituent: NP correspond to Thing-constituents, the PP to a Path-constituent, and the entire sentence to an Event. The converse of this correlation does not hold, e.g. many conceptual constituents of a sentence's meaning are completely contained within lexical items. The mapping between conceptual and syntactic categories is many-to-many but it is subject to markedness conditions. Each conceptual constituent has an argument structure feature, which allows for recursion of conceptual structure and hence an infinite class of possible concepts.

LOCALISM A second cross-categorial property of conceptual structure goes back to the *localistic* theory. The formalism for encoding concepts of spatial location and motion can be abstracted/generalized to many other semantic fields. Many verbs and prepositions appear in more semantic fields and in intuitively related paradigms.

Many implicative properties of verbs (such as *factive*, *implicative*, and *semifactive*) follow from generalized forms of inference rules developed to account for verbs of spatial motion and location. Each semantic field has its own particular inference patterns, e.g. in the spatial field, one fundamental principle stipulates that an object cannot be in two disjoint places at once. It follows that an object that travels from one place to another is not still in its original position. In the field of information transfer, this inference does not hold. A similar conceptual structure may apply to different parts of speech, as exemplified by the parallelism between iteration of actions and plural of things, or the bounded/unbounded distinction among verbs (event/process, telic/atelic) and the count/mass distinction among nouns.

PREFERENCE RULE SYSTEMS CS involves something similar to prototype theory or fuzzy set theory: Verbs have more "fuzzy truth conditions": climb = move up & grasp, see = gaze & realize. An event which satisfies both conditions at once, is more *stereotypical*. An example from an other part of speech is nouns that denote form and function as two conditions (e.g. *book*). When one lacks information about the satisfaction of the conditions, they are invariably assumed to be satisfied as default values.

2.2.5.1 *Argument Structure and Thematic Roles*

THE STATUS OF THEMATIC ROLES CS has a notion of thematic roles which has greatly influenced `4lang`. In Jackendoff (1990, Section 2.2)'s approach, thematic roles are structural configurations in CS.

```
DO(John,CAUSE(HAVE(Bill,book)))
```

E.g. the traditional Source/Goal, "the object from/to which motion proceeds", can be structurally defined as the argument of the Path-function FROM/TO. Agent is the first argument of the Event-function CAUSE, and Experiencer is an argument of some function having to do with mental states. A list of a verb's arguments can be constructed simply by extracting the indices from the verb's lexical conceptual structure. The hierarchy of thematic roles is "cca. provided" by the relative depth of embedding of the indices in conceptual structure. The CS account of thematic roles combines semantic intuitions with a rich system governed by its own combinatorial properties. Each kind of argument position plays a distinct role in rules of inference. Not only NPs receive thematic roles. For instance, *green* is a Goal in *The light changed from red to green.*, and *shut up* is a Goal in *Bill talked Harry into shutting up.*, not the thematic role for a subordinate clause, as suggested in Lexical Functional Grammar. Clauses can occur in various thematic roles, just as Things can. There's no "default" thematic role in the sense that Objective is "default" or "neutral" in (Fillmore 1968): in CS, an NP must correspond to a specific argument position in conceptual structure and therefore must have a specific thematic role. Even *Theme* or *Patient*, which have been taken to be such a default role, have a specific structural definition.

ARGUMENT FUSION AND SELECTIONAL RESTRICTIONS In CS, and similarly in `4lang`, a verb's lexical representation can include information about a participant which is not even syntactically expressed. In order for a sentence to be understood, this fine CS must exist. Selectional restrictions are explicit information that the verb supplies about its arguments. Formally, they correspond to the conceptual structure that occurs within an indexed conceptual constituent.

CS is a unification-based system: if two conceptual structures contain incompatible information, (if the offending features are sisters in a taxonomy of mutually exclusive possibilities, such as Thing/Property/Place/Event/etc. or solid/liquid/gas) their fusion is anomalous. `4lang` in does not implement such hard constraints.

E.g. drink vs butter both mean "cause something to go someplace". They differ semantically in what they stipulate about the Theme and the Path. The direct object of *butter* is the Goal, and the Theme is completely specified by the verb, while the direct object of *drink* is the Theme, and the Path is (almost) completely specified by the verb. It is part of the meaning of *order* that the recipient (or Goal) of an order is under obligation to perform the action described by the complement clause, and that of *promise* that the issuer (or Source) of a promise undertakes an obligation to perform the action described by the complement.

Both CS and `4lang` have many ways of expressing conceptual structure within arguments of the verb (which is part of the verb's meaning);

positions of indices (in `4lang` terms, deep cases, see Chapter 6), i.e. the way the verb links its arguments to syntactic structure; and selectional restriction and implicit arguments.

MULTIPLE THEMATIC ROLES FOR A SINGLE NP Jackendoff (1990, Chapter 3) investigates the q-Criterion, i.e. that each subcategorized NP (plus the subject) corresponds to exactly one argument position in conceptual structure, and that each open argument position in conceptual structure is expressed by exactly one NP. In Jackendoff's view, the q-Criterion must be weakened, e.g. because of transaction verbs such as *buy, sell, exchange*, and *trade*, where there are two giving actions (that of the merchandise and the money), and the seller and the buyer have two semantic roles apiece; or *chase*, where both Agent and Patient move. We will see that deep cases in `4lang` are less semantic: *buy* has an agentive subject and the source, even if the latter gives money voluntarily. In contemporary computational systems, we can assume a sentence analyzed for syntactic dependency, and the task of deep cases is to mediate between the dependency annotation and semantic representation.

UNIFYING LEXICAL ENTRIES (Chapter 4) investigates argument structure alternations that can be captured by the same lexical entry. `4lang` goes the same path. Potential modifiers (of place, time, and manner) are not encoded anywhere in the lexical entry. The problem of causatives *(The box slid/Bill slid the box down the stairs)* is solved in the Unaccusative Hypothesis fashion we will discuss in Chapter 6.

A more special example is *climb* with three syntactic frames: null complement, direct object or PP. CS wants to account for the difference that only the null entails that the subject reaches the top. `4lang` disregards such differences, not in order to codify such a coarse level of mental representation, but as an engineering shortcut. More productive lexical processes e.g. passive participles from verbs can be expressed in terms of manipulations on the argument indices. In `4lang`, passives are already handled by the dependency parse. Jackendoff also discusses verbs with some spatial feature in their meaning (*point, surround, cover, support*) which would go beyond the limits of the present thesis.

SOME FURTHER CONCEPTUAL FUNCTIONS Jackendoff (1990, Section 5.2) investigates verbs of manner of motion like *curl, writhe*, or *dance*. These are less interesting for our present purposes, as the working method of `4lang` is to define manually only some defining vocabulary, which can be used to define all other words automatically.

While this topic is beyond the scope of the present thesis, we quote some ideas by Jackendoff on *conceptual clause modification*. Jackendoff offers a partial taxonomy of functions that convert a State or Event

into a restrictive modifier of another State or Event (syntactically: subordinating conjunctions that turn sentences into restrictive modifiers).

- Cause (why?) has logically two types: reason, `FROM`, a variety of the usual `FROM`; and purpose, goal, or rationale (the intention may be the speaker's or attributed to the Agent), `FOR`, a variety of `TO` or `TOWARD`.
- In accompaniment *(Bill came with Harry)* there is a mutual dependence between Bill's coming and Harry's, and Bill is "foregrounded". This asymmetrical relation is "more than conjunction but less than causation".
- Exchange, reward or punishment is a voluntary act of social cognition, based on assessment in legal and economic systems, which is worth a separate satus in cognitive semantics.

More of these subordinators are similar to spatial functions both in morphology and the inferences associated with them. Cross-linguistic study is important here, of course: if the same apparently idiosyncratic fact appears in language after language, something is being missed. Conversely, if an apparently principled English fact is violated in other languages, the principle must be questioned.

SOME FEATURAL ELABORATIONS OF SPATIAL FUNCTIONS Jackendoff aims at a featural decomposition of verb meaning. E.g. he introduces a feature opposition in spatial location, say Location versus Contact (or ±contact) which is present in the prepositional system, where *on* and *against* contrast with *in, next to, alongside, above*, and in the lexicon, where *stroke*, *scratch rub*, and *brush* specify motion while in continuous contact with the object. `4lang`, in contrast, tries to capture words with other words instead of features.

THE ACTION TIER AND THE ANALYSIS OF CAUSATION Jackendoff (1990, Section 7.1) decomposes thematic roles to two dimensions: The Action Tier distinguishes the Actor and Patient, while the thematic tier (Theme, Source, and Goal) deals with motion and location. Thus *What happened to Pat?* or *What did Agt do to Pat?* is orthogonal to *What moves where?*.

2.2.6 *English Verb Classes and Alternations*

As another source of semantic knowledge, Levin (1993) points out that the expression and interpretation of arguments is to a large extent determined by the verb's meaning. The introduction of the book exemplifies this with *break, cut, hit,* and *touch.* Each verb shows a distinct pattern with respect to three alternations, the middle alternation (*This bread*

cuts easily.), the conative construction (*cut at*), and the body-part alternation (*Margaret cut Bill on the arm.*). There are other verbs that show the same pattern of behavior: Break Verbs: *break, crack, rip, shatter, snap*, Cut Verbs: *cut, hack, saw, scratch, slash*, Touch Verbs: *pat, stroke, tickle, touch*, and Hit Verbs: *bash, hit, kick, pound, tap, whack.*

Levin's analysis is based on relevant meaning components. The body-part possessor ascension alternation needs 'contact', while the conative alternation: needs both 'motion' and 'contact'. *Touch* is a pure verb of contact, *hit* is a verb of contact by motion, *cut* is a verb of causing a change of state by moving something into contact, and *break* is a pure verb of change of state. This explains which verb participates in which alternation.

These phenomena are manifested across languages by verbs of the same semantic types. To the extent that languages are similar, the same meaning components – and hence the same classes of verbs – figure in the statement of regularities concerning the expression of arguments. The classes have in common a range of properties, including the possible expression and interpretation of their arguments, and the existence of certain morphologically related forms.

The meaning component analysis is related to "semantic bootstrapping" models of child language acquisition built on the assumption that a word's syntactic properties are predictable from its meaning. Meaning components identified via the study of semantic/syntactic correlation show considerable overlap with those posited in language acquisition.

Levin investigates intricate and extensive patterns of syntactic behavior: subcategorization frame of a verb, diathesis alternations, morphological properties and extended meanings.

Part I of the book introduces diathesis alternations that are relevant to lexical knowledge, subdivided into groups on the basis of the syntactic frames involved: transitivity alternations, alternate expressions of arguments (mostly within the verb phrase), alternations that permit "oblique" subjects, and a variety of other types. Part II presents a large number of semantically coherent classes of verbs[3]. Levin tries to strike a balance between breadth and depth of coverage. He ignores verbs taking sentential complements except when they show interesting behavior with NP or PP complements; verbs derived by productive morphological processes, such as zero-derivation, prefixation (*un-*, *de-*, *dis-*, *re-*,

3 Put; Remove; Send and Carry; Exert Force: Push/Pull; Change of Possession; Contribute; Learn; Hold and Keep; Concealment; Throw; Contact by Impact; Hit; Poke; Contact: Touch; Cut; Combine and Attach; Separate and Disassemble; Color; Image Creation; Illustrate; Creation and Transformation; Engender; Calve; Verbs with Predicative Complements; Perception; Psych-Verbs (Psychological State); Desire; Judgment; Assessment; Search; Social Interaction; Communication; Sounds Made by Animals; Ingest; Involve the Body; Groom and Bodily Care; Kill; Emission; Destroy; Change of State; Lodge; Existence; Appearance, Disappearance, and Occurrence; Body-Internal Motion; Assume a Position; Motion; Avoid; Linger and Rush; Measure; Aspectual Verbs; Weekend; Weather

etc.) or suffixation (*-ify*, *-ize*, *-en*, etc.); and inherent lexical aspect of verbs (aktionsart). It is left as an open research question whether a complete *hierarchical* organization of English verb *classes* is possible or even desirable.

2.2.7 *The generative lexicon*

Traditional lexicons and WordNet (Section 2.4.1) have very fine-grained sense distinctions, and the relations between different senses are mostly not represented. Pustejovsky (1995) call these resources *sense enumeration lexicons (SELs)* , and proposes the *generative lexicon (GL)* as an alternative, where lexemes have richer structure, and the virtually infinite semantic types a lexeme may have arise in context, by co-composition with the similarly flexible representations of other words, similarly to how infinitely many sentences are generated from a finite lexicon by recursive generative grammars in syntax. While the core of the GL is organized among semantic types, and is thus less interesting in the context of `4lang`, the theory has many features worth studying from a more association-based point of view as well.

GL builds in a classification of word polysemy to homonymy and polysemy proper, or, in Weinreich (1964)'s terms, contrastive and complementary ambiguity. *Contrastive ambiguity (i.e. homonymy)* is the coincidence of unrelated meanings, while *complementary ambiguity (polysemy)* refers to logically related word senses, manifestations of the same basic meaning in different contexts, possibly different POSs. Whether homonymic senses are historically related or accidents of orthographic and phonological blending, is largely irrelevant for purposes of lexicon construction and the synchronic study of meaning. The two types of ambiguity also differ in whether the disambiguation of co-occurring words help each other: contrastive disambiguation works so that once the context or domain for one item has been identified, the ambiguity of the other items is also constrained (contextual priming). This does not hold for sense narrowing in complementary ambiguity, where one sense may be entailed by the other sense. Pustejovsky mentions classes of complementary polysemy where the senses correspond to different semantic types like Count/Mass (*lamb*), Container/Containee (*bottle*), Gap/Frame (*door, window*), Product/Producer (*newspaper, Honda*), Plant/Food (*fig, apple*), Process/Result (*examination, merger*), Place/People (*city, New York*), and Change-state/Create (*bake*).

Pustejovsky 1995, sec 3 goes further to define *logical polysemy* as a complementary ambiguity where there is no change in lexical category, and the multiple senses of the word have overlapping, dependent, or shared meanings.

Pustejovsky lists three arguments showing the inadequacies of SELs for semantic description and that to maintain compositionality, we must enrich the representations of the lexical items:

- *The Creative Use of Words*, that words assume new senses in novel contexts,
- *The Permeability of Word Senses*, that Word senses are not atomic definitions but overlap and make reference to other senses of the word, and
- *The Expression of Multiple Syntactic Forms*, that a single word sense can have multiple syntactic realizations.

GL involves four levels of lexical representation: argument, event, qualia, and inheritance structure.

Argument structure specifies the number and type (semantic and syntactic) of arguments a predicate takes. This is by far the best understood of the four levels in generative linguistics (e.g. Chomsky's Theta-Criterion, Lexical Functional Grammar (Bresnan 1978, 2001)), and argument structure is also the strongest determinant or constraint on the acquisition of verb meaning by children. Pustejovsky distinguishes four types of arguments (illustrated for verbs), *true* obligatory arguments subject to the theta criterion; *default* arguments that are necessary for the logical well-formedness of the sentence, but may be left unexpressed on the surface; *shadow* arguments e.g. incorporated semantic content (the instrument of *kick* or *butter*); and *(true) adjuncts* that are associated with verb classes and not with the representation individual verbs, including temporal or spatial modifiers. The categorization of arguments induces one of verb alternations as well: those that result in the expression of true arguments versus which involve the expression of optional ones.

Event structure is for the representation of information related to Aktionsarten and event type, in the sense of Vendler (1967): event type (state, process, and transition) and subeventual structure. Besides the relation between an event and its subevents, GL involves overlap and inclusion of subevents as well, and one of the subevents may be the *head* of the event. In `4lang`, there are three potential subevents, the unmarked (present) one, the one connected by the binary concept `after`, and that connected by `before`.

Qualia structure is the set of properties or events associated with a lexical item which best explain what that word means, such as its constituent parts, purpose and function, mode of creation, etc. More formally, these aspects are

- CONSTitutive, the relation between an object and its constituent parts, e.g. "text in a novel is characteristically a narrative or story, while a dictionary is by definition a listing of words", its material, and also what this object is part of.
- FORMAL: orientation, magnitude, shape, dimensionality, color, position;

$$\begin{bmatrix} \textit{novel} & \\ \text{QUALIA} = & \begin{bmatrix} \text{CONST} & = \textit{narrative} \\ \text{FORMAL} & = \textit{book} \\ \text{TELIC} & = \textit{reading} \\ \text{AGENT} & = \textit{writing} \end{bmatrix} \end{bmatrix}$$

Figure 4: Qualia structure of *novel*.

- TELIC: purpose and function, how we use a thing, or the purpose that an agent has in performing an act. Direct TELIC, e.g. beer is made in order that it will be drunk, is distinguished from instruments e.g. knives are made to cut with them; and finally,
- AGENTive specifies how they come into being, a mode of explanation that will distinguish natural kinds from artifact (e.g. cookies, cakes, and bread are typically baked);

see Figure 4. The model is inspired in part by Moravcsik (1975)'s interpretation of Aristotle's modes of explanations. Instead of describing the GL formalism (which is reminiscent of HPSG) we concentrate on the semantic content. The qualia structure plays an important role in how we understand sentences, e.g. by knowing that the TELIC of *movie* is *watch*, we understand *John enjoyed the movie* so that he enjoyed *watching* it.

The last one of the four levels, inheritance identifies how a lexical structure is related to other structures in the type lattice.

The four levels are connected by generative devices providing for the compositional interpretation of words in context. Though GL is a strongly typed model, and these devices (e.g. type coercion and shifting, selective binding and co-composition) play the role of fitting items in novel type environments, the basic idea that predicate-argument binding can refer to subevents in the semantic representation is a feature shared with `4lang`.

While Pustejovsky criticizes the lexical semantic literature for overemphasizing the role of verbs, their classes and alternations, he also devotes a chapter to this topic, more concretely causation. The point is that members of alternations, e.g. the transitive and the intransitive variant of a verb are generated from the same item in GL.

2.3 EARLY RESOURCES

2.3.1 *Roget*

Now we turn to lexical resources in NLP, starting with Roget's thesaurus which remained a strong baseline 87 years after its creation

Hierarchy	1911	1987
Class	8	8
Section	39	39
Subsection	97	95
Head Group	625	596
Head	1044	990
Part-of-speech	3934	3220
Paragraph	10244	6443
Semicolon Group	43196	59915
Total Words	98924	225124
Unique Words	59768	100470

Table 3: Comparison of the 1911 and 1987 editions of the Roget's Thesauri by Kennedy and Szpakowicz (2008)

(Kennedy and Szpakowicz 2008). `4lang` uses the thesaurus to operationalize (Kornai 2019, Section 6.4) the concept of a semantic field (Trier 1931), a set of "conceptually related terms that are likely candidates to be defined in terms of one another such as color terms, legal terms" or naive physics, as we have seen in Section 2.1.6.

Roget's organizes words in a hierarchy. Kennedy and Szpakowicz (2008, 2014) compares the 1911 and 1987 versions of Roget's to each other and to WordNet (see Section 2.4.1). The older version of Roget is in the open domain along with related NLP-oriented software packages. Data size in each level of the hierarchy is shown in Table 3. Units near the leaves of the hierarchy are more specific than those close to the root.

There are nine levels in Roget's Thesaurus hierarchy (including words themselves):

- Class, e.g. "Words Expressing Abstract Relations", a
- Section in that Class is "Quantity" with a
- Subsection "Comparative Quantity".
- Heads can be thought of as the heart of the Thesaurus.
- The Paragraph and Semicolon Group are not given names, but can often be represented by the first word. The closest counterpart of WordNet's synsets is the Semicolon Group, which usually contains near-synonyms. Division by part-of-speech is quite low in Roget's hierarchy, not at the very top as in WordNet.

Kennedy and Szpakowicz (2008) conduct experiments in synonym identification and word relatedness. The idea for determining semantic relatedness between pairs of terms is that terms that appear closer

together in the Thesaurus get higher weights than those farther apart. The authors find that while the 1987 version of the Thesaurus is better, the 1911 version performs surprisingly well and often the differences between the versions of Roget's and WordNet are not statistically significant.

Kennedy and Szpakowicz (2014) evaluate automatic updates of Roget's Thesaurus in the selection of the best synonym from a set of candidates, pseudo-word-sense disambiguation, and SAT-style analogy problems. They use the resource to learn how to place new words in the correct locations in the same hierarchy.

2.3.2 KL-ONE

One primary function of semantic networks, as we have already seen in more examples, is inheritance and some formalization of the genus versus differentia specifica. Another system that has impact in this regard on `4lang`, at least in terminology, is KL-ONE (Brachman and Levesque 1985). KL-ONE *concepts* are described by their subsuming concepts (their super-concepts) and their local internal structure expressed in *roles* (which describe relationships like properties or parts) and structural descriptions, which express the interrelations among the roles.

> A Concept must have more than one super-concept (if there are no local restrictions), differ from its superconcept in at least one restriction, *or* be primitive. A Concept with no local restrictions is defined as the conjunction of its super-concepts. Superconcept serves as a proximate genus, whereas the local internal structure expresses *essential differences*, as in classical classificatory definition (Sellars 1917). The network structure formed by the subsumption relationships between Concepts [is] a *taxonomy*. (Emphasis added)

KL-ONE instigated first-class status for roles that we will briefly discuss in Chapter 6. We conclude this description of KL-ONE with *contexts*. Individual concepts, that uniquely describe individuals, are associated to some context. Assertions about co-reference and existence are also always relative to some context so as not to affect the taxonomy of generic knowledge. Context provides the mechanism for reasoning about hypotheticals, beliefs, and desires.

2.3.3 *Cyc*

Most of the problems discussed in Section 2.1.4 have not been solved to this day, and though expert systems in specific domains brought a second summer, the second winter also arrived due to brittleness

outside theses narrow domains. The drop in reputation and funding was a sign of the need to represent the wisdom of a kindergarten child in a knowledge base (KB). Cyc (Lenat and Guha 1990) is an early example of effort in this direction. In retrospect, their success lies between the two extremes they formulated as at least providing some insight into issues involved in ontology population with "an indication as to whether the symbolic paradigm is flawed" and the more optimistic one that "no one in the early twenty-first century even considers buying a machine without common sense". For `4lang`, Cyc is relevant especially for the status of primitives, see Section 3.2.

Lenat and Guha (1990) organize their paper along the three tasks in building a KB: the (logical) language (CycL), the procedures for manipulating knowledge, and populating the KB. Though the authors frame understanding as including "beliefs, knowledge of others' [...] limited awareness of what we know, various ways of representing things, [and] knowledge of which approximations (micro-theories) are reasonable in various contexts", they note that an ontology is required already for word sense disambiguation, anaphora resolution, or understanding ellipsis. In our description of Cyc we concentrate on its aspects with the greatest impact on `4lang`, the language and the database, rather then inference.

The two systems are already similar in their methodologies: the core of the Cyc ontology was built manually and later they crossed to primarily automatic knowledge entry via natural language understanding in the 80s.

> We developed our representation language incrementally as we progressed with [the task of knowledge encoding]. Each time we encountered something that needed saying but was awkward or impossible to represent, we augmented the language to handle it. Every year or two we paused and smoothed out the inevitable recent "patchwork."

The language is summarized as frame-based and embedded in a more expressive predicate calculus framework along with features like representing defaults or reification (allowing one to talk about propositions in the KB). As for the inference machine, they abandon the AI tradition of a single, very general mechanism (e.g., resolution) for problem solving and prefer special data structures and algorithms for problems of varying complexity as done in traditional computer science.

The main difference between the Cyc KB and `4lang` is that we concentrate on the core vocabulary, while this distinction is not made in the Cyc KB where, though many of the one or two million assertions are general rules, some are specific facts dealing with particular entities and events (e.g. famous people and battles.)

A great heritage of `4lang` from Cyc is the use of non-monotonic reasoning: most assertions are *default* beliefs and the addition of new

facts can cause them to be retracted. Cyc is also similar to present day question answering systems that inference is based upon a (quickly identified) small subset of relevant sentences.

Though Cyc is *strongly typed* (as opposed to the type-free `4lang`), it offers us many insights. Lenat and Guha frequently use "set-theoretic notions to talk about collections, but these collections are more akin to what W. Quine (1969) termed *natural kinds*, e.g. *dog* or *lemon*, that are usually assumed not to be completely definable as intersections of more primitive classes. Collections are organized in a generalization-specialization hierarchy" (Brachman and Levesque 1985).

Cyc handles *time and actions* analogously to space: time and events are substances. "One could take a glob of peanut butter and separate out all the peanut chunks, and these alone do not form a glob of peanut butter. [...] The substancehood principle applies only to pieces larger than the *granule* of that substance." 'Walking' is a type of temporal substance by the same token.

As there are "orthogonal ways of breaking down a physical object, there [are two] orthogonal ways of breaking down an action:" actors and subEvents. There are separate categories of slots that are used in order to relate actors to actions and subEvents to events. To put it so simply that may seem brutal from a strongly typed point-of-view, but excellent for `4lang` purposes: actor slots are roles like *performer, victim, and instrument* and sub-event slots are 'before', 'during' and 'after' the action. The later are the predecessors of the `4lang` concepts representing event structure with the same names (except for 'during' which is unmarked in `4lang`), and can also be compared to the PRE- and POST- procedures (conditional execution) and WHEN- (side effects) in KL-ONE(see the previous section).

2.3.4 *Computational lexicography for NLP*

Now we reach the dawn of corpus linguistics, and Boguraev and Briscoe (1989, Introduction) draws the attention to lexical resources, their theoretical role and applications in traditional linguistics and NLP-based systems. This book analyses The Oxford Advanced Learner's Dictionary of Current English (LDOCE), which is important for `4lang`, because our hand-written definitions heavily relied on it. Traditional lexicons contain tens or hundreds of thousands of lexical items, and computational lexicography and lexicology have developed disciplines with their own workshops and conferences. While NLP has established new lexical knowledge bases (KBs) for a wide variety of researchers and applications, reusing existing lexical resources offers further room for improvement. While machine readable dictionaries (MRDs) represent a considerable tradition where much work has already been done, difficulties arise because these resources are produced for human use, and

may make inconvenient assumptions, and rely on the users' linguistic and common sense knowledge which machines do not have.

The book has made a great influence on `4lang`, as both lines of research strive to make information in MRDs accessible for machine use, and with this information, evaluate and improve computational semantics systems and linguistic theories. Three decades since Boguraev and Briscoe (1989) have proven that lexicons derived from MRDs for use by machine are different from conventional dictionaries in how they organize and represent information, but the same dictionary database (DB) can be used for both automated and human use. Some reoccurring themes of the book are the division between lexical semantics and pragmatic knowledge, the border between rules and the lexicon, and the acquisition of POS and subcategorization information with syntactic features.

2.3.4.1 *The nature of a dictionary entry*

In Boguraev and Briscoe (1989)'s view on the lexicon vs rules division, a general-purpose dictionary DB should be as inclusive and theoretically uncommitted as possible. E.g either one assumes a rule of *re-* prefixation or one needs to list elements like *reissue, reclaim* and *repay.*

The entries in most dictionaries distinguish 'homographs' of a word form when it serves as noun, verb or some other POS. Entries start with the form (headword, spelling, hyphenation, phonetics variants, allomorphs, stress) and information on the distributional behaviour (either with a simple word class tag, e.g. The Collins English Dictionary, or with elaborate subcategorization information, e.g. The Oxford Advanced Learner's Dictionary of Current English, LDOCE, or The Collins COBUILD English Language Dictionary).

Regarding the meaning, dictionaries tend to provide definition(s), examples, cross references; grammar and stylistics of usage; synonyms, antonyms, related words; picture, etymology; and derived words, compound terms, idiomatic or common phrases, expressions and collocations. LDOCE also provides semantic notions in the form of so called *subject* and *box* codes, which specify the semantic field (e.g. politics, religion, language) and selectional restrictions (e.g. *sandwich* prefers an abstract or human subject). The language of dictionary definitions tends to be of a restricted form. In LDOCE, the vocabulary is restricted to approximately 2200 words used mainly in their most common sense, which theoretically would cut down circularities (but see the next paragraph). Unfortunately derivational morphology is applied to these words in a rather liberal way. Representation is made difficult by the fact that there is a continuum between the minimal semantic knowledge implied by the use of a particular word (word sense) and the special (or expert) knowledge relevant to its use in a *domain* context.

2.3.4.2 *Reliability and utility of MRDs*

The preface to the published version of LDV claims that 'a rigorous set of principles was established to ensure that only the most 'central' meanings of [a controlled vocabulary of] 2000 words, and only easily understood derivatives, were used'. 'Body' is part of the definitional vocabulary and has as its central (1) meaning "the whole of a person". However, Boguraev and Briscoe (1989) point out that *parliament* is defined as "a law-making body", utilizing the meaning of body (5) "a number of people who do something together". To make things worse, about 30 non-LDV words are used in definitions, e.g. *aircraft* is used 267 times.

Besides the already mentioned liberal use of derivatives ('container' is used for the definition of box2(1), even though only the verb *contain* is considered to be primitive), circularity (*container* ⇔ *box*) also arises. Another related problem is the use of phrasal verbs made up from verbs and particles taken from the restricted vocabulary but, of course, with a non-compositional meaning.

In an other chapter, Vossen, Meijs, and Broeder (1989) derive a syntactic typology for the structures of the meaning descriptions of each of the major parts of speech (POS) in a dictionary. The typology combines hyponyms and adjectives, with subject field, speech register, and sociolect codes.

2.3.4.3 *Connectionism, word ambiguity, and knowledge*

The final chapter (Wilks et al. 1989) investigates the relation between connectionism and word ambiguity. The authors of the thirty-year-old paper realize that connectionism shares properties with compositional semantics, and they do not expect to distinguish representations for particular word senses, but to be simply different aspects of a single non-symbolic representation, and to correspond (if to anything) to a selection of different weighted arcs. They advocate weighted symbolic representations. This view applies to issues of word sense for compositional semantics (discreteness of word senses vs. continuity and vagueness).

The position in the chapter is that the inseparability of knowledge and language goes far, and knowledge for certain purposes should be stored in text-like forms. The authors compare the semantic structure of dictionaries to the underlying organization of knowledge representations, and observe similarities: computational semantics converges with knowledge acquisition and computational lexicography. The chapter investigates whether it is right to assume word 'sense', direct from traditional lexicography and MRDs. (The answer is yes.) Another question is whether a dictionary is a strong enough *knowledge base*. Not directly, but its content can be made explicit by additional information.

Collecting the initial information (bootstrapping) is needed from the dictionary itself or some external resource.

2.4 MODERN LEXICAL RESOURCES

> Usage seems to be inversely proportional to representational complexity. — *(Russell and Norvig 2002)*

The final section of this chapter introduces modern lexical resources, which serve as the basis of any kind of supervised NLP research. Every experiment reported in the main part of the thesis relied on one of them.

2.4.1 *WordNet*

As a lexical NLP resource, the baseline for comparing `4lang` to is WordNet: the English (Princeton) WordNet (Miller 1995) and probably the Hungarian edition (Miháltz et al. 2008)[4] as well. WordNet follows the lexicographic tradition of treating POSs separately, and words are grouped by semantic equivalence to 117 000 *synsets* with a definition ("gloss") each and in most of the cases, sentences illustrating the use of the words in the set. WordNet disambiguates word forms to many senses (synsets) to account for fine distinctions in their usage. This opposes to the monosemic approach `4lang` follows. An aspect of WordNet which is more instructive for `4lang` is a variety of binary relations.

2.4.2 *Frame semantics and FrameNet*

Jurafsky (2014) introduces *frames* as a rather general representation that expresses the background contexts or perspectives by which a word or a case role could be defined. The name came from the pre-transformationalist (1974) view of sentence structure as consisting of a frame and a substitution list. Frames were also called *scripts* or *schemata*.

In Kornai (2008, Section 5.3)'s reflection, the original intention was to use scripts as repositories of commonsense procedural knowledge: what to do in a restaurant, what happens during a marriage ceremony, etc.; represent the actors fulfilling roles, e.g. that of the waiter or the best man; and decompose the prototypical action in a series of more elementary sub-scripts such as 'presenting the menu' or 'giving the bride away'. Kornai relates scripts to "linguistically better motivated models", in particular discourse representation theory, whose scope is more modest, being concerned primarily with the introduction of new entities (the owner, the best man). Scripts have also influenced studies of ritual.

4 https://github.com/dlt-rilmta/huwn

Turning to Jurafsky (2014)'s account of verbal case frames, Fillmore was also inspired by lists of slots and fillers used by early information extraction systems, but his version of this idea was more linguistic. The motivating example was the Commercial Event frame *(buy, sell, cost, pay, charge)*. Frames could represent perspectives on events, e.g. *sell* vs *pay*. Alternative senses of the same word might come from their drawing on different frames. The perspective-taking aspect of frame semantics influenced framing in linguistics and politics.

FrameNet (Baker, Fillmore, and Lowe 1998) combined Fillmore's early ideas on semantic roles with his later work on frames and his interest in corpus lexicography. Fillmore compiled a large set of frames, each of which consisted of

- lists of constitutive roles or "frame elements": sets of words that evoke the frame,
- grammatical information expressing how each frame element is realized, and
- semantic relations between frames and between frame elements.

Palmer, Gildea, and Xue (2010, Section 1)'s introduction to linguistic theories and semantic representations "ends where it began, with Charles Fillmore". In this and the following two sections, we introduce a couple of verb-related resources. Palmer, Gildea, and Xue (2010, Section 2) describes these resources as having differing goals, and yet being surprisingly compatible. They differ primarily in the granularity of the semantic role labels. FrameNet labels the arguments of *approve* as Grantor and Action. PropBank uses very generic labels such as Arg0, Arg1, ... VerbNet, on the third hand, has several alternative syntactic frames and a set of semantic predicates. VerbNet marks the PropBank Arg0 as an Agent and the Arg1 as a Theme. The three resources can be seen as complementary.

Based on Filmore's Frame Semantics, FrameNet (Baker, Fillmore, and Lowe 1998) describes a particular situation or event along with its participants. Semantic roles are called *Frame Elements (FE)*, and they are defined for each semantic frame. The predicate is called *Lexical Unit (LU)*. All LUs in a semantic frame share the same set of FEs. FEs are fine-grained semantic role labels, e.g. the Apply-heat Frame includes a Cook, Food, and a Heating Instrument.

A frame can also have adjectives and nouns such as nominalizations. FEs are classified in terms of how central they are: core (conceptually necessary for the Frame, roughly similar to syntactically obligatory), peripheral (such as time and place; roughly similar to adjuncts) or extra-thematic (not specific to the frame and not standard adjuncts but situating the frame with respect to a broader context).

Lexical items are grouped together without consideration of similarity of syntactic behavior, resulting in rich, idiosyncratic descriptions.

E.g. *buy* and *sell* both belong to the semantic frame 'Commerce_buy', which involves a Buyer and Seller exchanging Money and Goods. Buyer and Goods are core FEs for this frame while Seller and Money are Non-Core FEs. Other Non-Core FEs include but are not limited to Duration (the length of time the Goods are in the Buyer's possession), Manner, Means, Place, Rate, and Unit, the unit of measure for the Goods.

2.4.3 *VerbNet*

VerbNet (Kipper et al. 2008) is midway between PropBank and FrameNet in lexical specificity, but it is more similar to PropBank with its close ties to syntactic structure. VerbNet consists of hierarchically arranged verb classes, extended from the Levin classes (see Section 2.2.6): Levin has 240 classes, with 47 top level classes and 193 second and third level. Original Levin classes constitute the first few levels in the VerbNet hierarchy, with each class subsequently refined. VerbNet has added almost 1000 lemmas as well as 200 more classes. There is now a 4th level of classes and several additional classes at the other 3 levels.

VerbNet adds to each Levin class an abstract representation of the syntactic frames with explicit correspondences between syntactic positions and the semantic roles (e.g. *break*: Agent REL Patient, or Patient REL *into pieces*). Argument list consists of semantic roles (Agent, Patient, Theme, Experiencer, etc., 24 in total), and selectional restrictions on the arguments that are expressed using binary predicates that describe the participants during *stages of the event.*

VerbNet has class-specific interpretations of the semantic roles; 3,965 verb lexemes with 471 classes; links to similar entries in WordNet, OntoNotes groupings, FrameNet, and PropBank; and coherent syntactic and semantic characterization of the classes, which facilitate the acquisition of new class members.

Each VerbNet class contains a set of *syntactic frames.* Constructions such as transitive, intransitive, prepositional phrases, resultatives, and a large set of diathesis alternations listed by Levin are captured by semantic roles (such as Agent, Theme, and Location), the verb, other lexical items required for a construction or alternation, and semantic restrictions (such as animate, human, and organization). Syntactic Frames specify which prepositions are allowed, and the syntactic nature of the constituent (NP, PP, finite and nonfinite sentential complements).

Semantic predicates denote the relations between participants and events in the form of a conjunction of semantic predicates, such as motion, contact or cause, and START(e), END(e) or DURING(e) arguments to indicate when the semantic predicate is in force.

2.4.4 *PropBank*

PropBank consists of an annotated corpus (to be used as training data) and a lexicon. Semantic role labels are chosen to be quite generic and theory neutral, Arg0, Arg1, etc. The same semantic role is kept across syntactic variations. The lexicon lists, for each broad meaning of each annotated verb, its Frameset, i.e. the possible arguments in the predicate and their labels (its "roleset"), all possible syntactic realizations, and a set of verb-specific guidelines for annotators. PropBank is similar in nature to FrameNet and VerbNet although it is more coarse-grained, and more focused on literal meaning – as opposed to metaphorical usages and support verb constructions – than FrameNet.

PropBank defines semantic roles on a verb-by-verb basis:

- Arg0 is generally a prototypical Agent (Dowty 1991) while
- Arg1 is a prototypical Patient or Theme.
- There are no consistent generalizations for the higher numbered arguments, e.g. Arg2 can be beneficiary, goal, source, extent or cause.
- There are several more general ArgM (Argument Modifier) roles that can apply to any verb, and which are similar to adjuncts, e.g. LOCation, EXTent, ADVerbial, CAUse, TeMPoral, MaNneR, and DIRection.

These generic labels make high inter-annotator agreement possible. A roleset corresponds to a distinct usage of a verb. It is associated with a set of syntactic frames, the Frameset.

Three is a verb-specific descriptor field for each role, such as *baker* for 'Arg0' in *bake*, for use during annotation and as documentation, without any theoretical standing. The neutral, generic labels facilitate mapping between PropBank and other more fine-grained resources such VerbNet and FrameNet, as well as Lexical-Conceptual Structure or Prague Tectogrammatics.

Most role-sets have two to four numbered roles, but as much as six can appear, in particular for certain verbs of motion. PropBank lacks selectional restrictions, verb semantics, and inter-verb relationships.

Verb-Specific labels have their limitations. Inter-verb labels make inferences and generalizations based on role labels possible, because some encoded meaning is associated with each tag, which helps in training automatic semantic role labeling (SRL) systems. Researchers using PropBank as training data for the most part ignore the "verb-specific" nature of the labels, and instead build a single model for each numbered argument. This is feasible, because Arg0/Arg1 constitute 85% of the arguments, ArgMs are also labeled quite consistently. Arguments Arg2-Arg5 are highly overloaded, and performance drops significantly on them.

2.4.5 *ConceptNet*

As a basically word-level meaning representation framework, `4lang` has to be most relevantly compared to ConceptNet (Liu and Singh 2004). ConceptNet is knowledge graph, i.e. it connects words and phrases with labeled edges. It is designed to represent the general knowledge involved in understanding language in the form of relations between words such as 'A net is used for catching fish' '*Leaves* is a form of the word *leaf*' 'The word *cold* in English is *studený* in Czech', or 'O alimento é usado para comer' i.e. 'Food is used for eating'. This knowledge in version 5.5 (Speer, Chin, and Havasi 2017) has been collected from many sources that include expert created resources, crowd-sourcing, and games with a purpose.

The authors combine ConceptNet with word embeddings (Section 4.2) to get understanding that they would not acquire from distributional semantics alone, nor from narrower resources such as WordNet or DBPedia. The word embedding has been trained using a generalization of the *retrofitting* method (Faruqui et al. (2015), see Section 4.2.10). They demonstrated results on intrinsic evaluations of word relatedness, that was a popular way of evaluating word embeddings before the introduction of contextualized word representations (Section 4.3), and on applications of word vectors, including solving SAT-style analogies.

In the remainder of this section, we describe the ConceptNet representation based on Speer and Havasi (2012, Section 3). Assertions in ConceptNet can be seen as edges that connect its nodes, which are concepts (words or phrases). Assertions can be justified by other assertions, knowledge sources, or processes. Predicates (i.e. edge labels) can be interlingual relations, such as *IsA* or *UsedFor* (see Table 4); or automatically-extracted relations that are specific to a language, such as *is known for* or *is on*. Processes that read knowledge from free text, will produce relations that are not aligned with multilingual relations. In this case, the relation specifies the language and a normalized form, e.g. *A bassist performs in a jazz trio* translates to a `/c/en/perform_in` relation.

Negation in ConceptNet is a bit tricky. Conjunctions of assertions come with a positive or negative score, where a negative weight means we should conclude that the assertion is not true. The negation of such conjunction is not necessarily true either: It may in fact be a nonsensical or irrelevant. To represent a true negative statement, such as *Pigs cannot fly*, ConceptNet 5 uses negated relations such as `/r/NotCapableOf`.

2.4.6 *Deep Lexical Semantics*

Now we turn to Deep Lexical Semantics (Hobbs 2008), but motivate it from a more recent perspective. HellaSwag (Zellers et al. 2019) tests pre-trained deep language models like BERT by asking them which of

Relation	Sentence pattern
IsA	NP is a kind of NP.
UsedFor	NP is used for VP.
HasA	NP has NP.
CapableOf	NP can VP.
Desires	NP wants to VP.
CreatedBy	You make NP by VP.
PartOf	NP is part of NP.
Causes	The effect of VP is NP|VP.
HasFirstSubevent	The first thing you do when you VP is NP|VP.
AtLocation	Somewhere NP can be is NP.
HasProperty	NP is AP.
LocatedNear	You are likely to find NP near NP.
DefinedAs	NP is defined as NP.
SymbolOf	NP represents NP.
ReceivesAction	NP can be VP.
HasPrerequisite	NP|VP requires NP|VP.
MotivatedByGoal	You would VP because you want VP.
CausesDesire	NP would make you want to VP.
MadeOf	NP is made of NP.
HasSubevent	One of the things you do when you VP is NP|VP.
HasLastSubevent	The last thing you do when you VP is NP|VP.

Table 4: The interlingual relations in ConceptNet, with example sentence frames in English. Table from (Speer and Havasi 2012)

Composite Entities	perfect, empty, relative, secondary, similar, odd
Scales	step, degree, level, intensify, high, major, considerable
Events	constraint, secure, generate, fix, power, development
Space	grade, inside, lot, top, list, direction, turn, enlarge, long
Time	year, day, summer, recent, old, early, present, then, often
Cognition	imagination, horror, rely, remind, matter, estimate, idea
Communication	journal, poetry, announcement, gesture, charter
Persons	leisure, childhood, glance, cousin, jump
Microsocial	virtue, separate, friendly, married, company, name
Bio	breed, oak, shell, lion, eagle, shark, snail, fur, flock
Geo	storm, moon, pole, world, peak, site, sea, island
Material World	smoke, shell, stick, carbon, blue, burn, dry, tough
Artifacts	bell, button, van, shelf, machine, film, floor, glass, chair
Food	cheese, potato, milk, bread, cake, meat, beer, bake, spoil
Macrosocial	architecture, airport, headquarters, prosecution
Economic	import, money, policy, poverty, profit, venture, owe

Table 5: Concepts in Hobbs (2008)

the alternatives bellow finishes the short text *A woman is outside with a bucket and a dog. The dog is running around trying to avoid a bath. She...* the most appropriately.

1. rinses the bucket off with soap and blow dry the dog's head.
2. uses a hose to keep it from getting soapy.
3. gets the dog wet, then it runs away again.
4. gets into a bath tub with the dog.

The good answer is 3. Models struggle with this task. The authors note that while the wrong endings are on-topic, with words that relate to the context, humans consistently judge their meanings to be either incorrect or implausible. These problems suggest that for understanding, we need something beyond the meaning of the words, and their probability in different sentence contexts. We saw in Section 2.1.6 that Hayes (1979) suggested the construction of a formalization of a portion of common-sense knowledge about the everyday physical world along with a theory of meaning. Deep Lexical Semantics (Hobbs 2008) is a step in this direction.

Hobbs (2008) took a basic core of about 5000 most frequent synsets in WordNet; categorized these into sixteen broad categories, e.g. time, space, scalar notions, composite entities, and event structure; and sketched out the structure of some of the underlying abstract core theories of commonsense knowledge (see Table 5). The latter includes the basic predicates in terms of which the most common word senses need to be defined or characterized; axioms that link the word senses to the core

theories; and a kind of "advanced lexical decomposition", where the "primitives" into which words are "decomposed" are elements in coherently worked-out theories. Hobbs (2008) focuses on the 450 of these synsets that are concerned with *events* and their structure.

Hobbs has very similar principles to Hayes (1979): We must have underlying theories and axioms that link these to words. Concepts and axioms include domain-dependent knowledge, of course, but 70-80% of the words in most texts, even technical texts, are words in *ordinary* English. Hobbs chooses the core theory of *scales*, which will provide axioms involving predicates such as 'scale', '<', 'subscale', 'top', 'bottom', and 'at'. These are abstract notions that apply to partial orderings as diverse as heights, money, and degrees of happiness.

Some lexical and world knowledge can be acquired automatically, e.g. the correlation between "married" and "divorced", and maybe even the corresponding predicate-argument structures and, which way the implication goes and with what temporal constraints. But this is a too simple relation to axiomatize in comparison to the "range". In Hobbs's view, it is feasible to manually axiomatize the meanings of several thousand words, what can achieve the desired complexity and reliability of the core theories and the linking axioms.

Section 3 describes the following core theories that are crucial in characterizing *event* words:

- Eventualities and their Structure: states and events,
- Set Theory (modeled in a standard fashion),
- Composite Entities, including the predicate 'partOf' and the figure-ground relation 'at',
- Scales: partial orderings, monotone functions, the construction of composite scales, the characterization of qualitatively high and low regions of a scale (related to distributions and functionality), and constraints on vague scales,
- Change of State
- Cause. Recall that Hayes (1979) explicitly warned against trying to formalize causality, saying that what happens e.g. with liquids, is part of the liquids cluster, not part of some theory of 'what-happens-when'.

  In Hayes view, causality is characterized by two properties: If every eventuality in a causal complex happens, the effect happens; and everything in the causal complex is *relevant* to the effect in a way that can be made precise. Hobbs's approach to causality includes force-dynamic notions (Section 2.2.4) like *enable, prevent, help, obstruct, attempts, success, failure, ability*, and *difficulty*.

- Events. Changes of state and causality compose into more complex events, (conditional, iterative, cyclic, and periodic events). This part of the theory is linked with several well-developed ontologies for event structure.
- a well-developed theory of *time*,
- a rather sparse theory of *space*, and
- a large number of theories explicating a commonsense theory of *cognition*,
- the predicates 'possess' and 'remain' would be explicated in a commonsense theory of *economics*.

2.4.7 *Abstract Meaning Representation for Sembanking*

Now we turn to the most popular meaning representation framework, Abstract Meaning Representation (AMR, Banarescu et al. (2013)). The original paper illustrates the AMR method with a syntactic analogue. Syntactic treebanks have had tremendous impact on natural language processing. Whole sentence parsing unified separate tasks (e.g. base noun identification) and their evaluations. Now smaller tasks are naturally solved as a by-product of whole-sentence parsing, and in fact, solved better than when approached in isolation. By contrast, semantic annotation is balkanized with separate annotations for named entities, co-reference, semantic relations, discourse connectives, temporal entities, etc. Each annotation has its own associated evaluation, and training data is split across many resources. The idea behind AMR has been to unify the semantic landscape.

The authors wrote down the meanings of thousands of English sentences in simple, whole-sentence semantic structures. AMR and the tools associated with it have the following principles:

- Rooted, directed, edge-labeled, leaf-labeled graphs, easy for people to read, and for programs to traverse. This traditional format is equivalent to feature structures, conjunctions of logical triples, directed graphs, and PENMAN inputs. The latter is used for human reading and writing. The root of an AMR represents the focus of the sentence or phrase.
- AMR trees abstract away from syntactic idiosyncrasies, attempting to assign the same AMR to sentences that have the same basic meaning, e.g. *he described her as a genius*, *his description of her: genius*, and *she was a genius, according to his description* are assigned the same tree.
- Extensive use of PropBank framesets (see Section 2.4.4). For example, AMR represents `bond investor` using the frame `invest-01`, even though no verbs appear in the phrase.

- Agnostic about how to analyze/generate.
- Heavily biased towards English, not an interlingua.

AMR has a 50-page annotation guideline.

2.4.7.1 *AMR Content*

In neo-Davidsonian fashion, AMR introduces variables (or graph nodes) for entities, events, properties, and states. Leaves are labeled with concepts, so that `(b / boy)` refers to an instance (called b) of the concept 'boy'. Relations link entities, so that `(d / die-01 :location (p / park))` means there was a death d in the park p. When an entity plays multiple roles in a sentence, AMR employs re-entrancy in graph notation (nodes with multiple parents) or variable re-use in PENMAN notation.

Concepts are either English words (`boy`), PropBank framesets (`want-01`), or special keywords. The latter include special entity types (`date-entity`, `world-region`, etc.), quantities (`monetary-quantity`, `distance-quantity`, etc.), and logical conjunctions (`and`, etc.). There are approximately 100 relations:

- Frame arguments, following PropBank conventions. :arg0, :arg1, . . . , :arg5
- General semantic relations: :accompanier, :age, :beneficiary, :cause, :compared-to, :concession, :condition, :consist-of, :degree, :destination, :direction, :domain, :duration, :employed-by, :example, :extent, :frequency, :instrument, :li, :location, :manner, :medium, :mod, :mode, :name, :part, :path, :polarity, :poss, :purpose, :source, :subevent, :subset, :time, :topic, :value
- Relations for quantities. :scale, :quant, :unit,
- Relations for date-entities. :day, :month, :year, :weekday, :time,. . .
- Relations for lists. :op1, :op2, :op3, :op4, :op5, :op6, :op7, . . . , :op10
- The inverses of all relations, e.g. :arg0-of,
- Every relation has an associated reification, which is used when we want to modify the relation itself.

AMR's hundred relation types contrasts to `4lang`'s sparse inventory (in graphs, `4lang`uses 0-, 1-, and 2-arrows, see Section 3.1.3),

> but the difference between `4lang` and AMR is less severe than it may appear at first blush, since the overwhelming majority of AMR relations like `:employed-by` are simply treated as ordinary transitive predicates in `4lang` . . . Considerable

> technical differences remain, e.g. `4lang` does not countenance overt semantic passives like 'employed by'. (Kornai et al., manuscript)

The authors give examples of how AMR represents various linguistic phenomena. AMR handles some level of derivational morphology. Besides nominalizations that refer to a whole event or a role player in an event, *-ed* adjectives frequently invoke verb framesets, e.g. *acquainted with* and *-able* adjectives often invoke the AMR concept *possible*, but not always.

Most *prepositions* simply signal semantic frame elements, but they are kept if they carry additional information. Cases when neither PropBank nor AMR has an appropriate relation, e.g. *The man was sued in the case* are solved in like this:

```
(s / sue-01
:arg1 (m / man)
:prep-in (c / case))
```

NAMED ENTITIES Any concept in AMR can be modified with a :name relation. There are standardized forms for 80 named-entity types, e.g. person or country. Multiple forms of a concept are not normalized (*US* versus *United States*), and nor are semantic relations inside a named entity analyzed. This offers a uniform treatment to titles, appositives, and other constructions.

REIFICATION The sentence *The marble was not in the jar yesterday* is represented as

```
(b / be-located-at-91
:arg1 (m / marble)
:arg2 (j / jar)
:polarity -)
:time (y / yesterday))
```

If AMR would not use the reification, we would run into trouble, e.g.

```
(m / marble
:location (j / jar
:polarity -)
:time (y / yesterday))
```

cannot be distinguished from the representation of *yesterday's marble in the non-jar*. Some reifications are standard PropBank framesets (e.g., `cause-01` for :cause, or `age-01` for :age).

2.4.7.2 *Limitations of AMR*

AMR does not represent inflectional morphology and universal quantification, it does not distinguish between real events and hypothetical,

future, or imagined ones e.g. in `the boy wants to go`, `want-01` and `go01` have the same status, and noun compounds do not have a systematic representation, e.g. `history teacher` and `history professor` translate to `(p / person :arg0-of (t / teach-01 :arg1 (h / history)))` `(p / professor :mod (h / history))`, respectively, because `profess-01` is not an appropriate verb. It would be reasonable in such cases to use a NomBank (Meyers+ 2004) noun frame.

2.4.7.3 *Creating AMRs*

An AMR Editor allows rapid, incremental AMR construction. To assess inter-annotator agreement (IAA), as well as automatic AMR parsing, AMR developed the Smatch metric and associated script that measure the overlap between two AMRs by viewing each AMR as a conjunction of triples. Smatch takes the variable mapping that yields the highest F-score.

2.4.8 *Enhanced English Universal Dependencies*

`4lang` is a semantic model, and the division of labor principle suggests that a semantic project should defer the task of syntactic analysis to existing tools. Interfacing with syntax remains an important problem. Kovács et al. (2022) discusses how more recent `4lang` graphs are created from a Universal Dependencies (UD) representation created by Stanza (Qi et al. 2020). This section introduces recent development in syntactic analysis which is relevant for semantics. In creating so-called *enhanced++* English Universal Dependency graphs, Schuster and Manning (2016) are motivated by that many shallow natural language understanding tasks use dependency trees to extract relations between content words. They revisit and extend these dependency graph representations in light of the Universal Dependencies initiative and provide an enhanced and an enhanced++ English UD along with a converter from basic UD trees to enhanced and enhanced++ English UD graphs, which are part of Stanford CoreNLP and the Stanford Parser.

The authors point out that the usage of Stanford Dependencies (SD) representation has falls into two categories: syntactic and a shallow semantic representations. Syntactic tasks proper, such as source-side reordering for machine translation or sentence compression, require a syntactic tree: a sound syntactic representation is more important than the relations between individual words. These trees need to be strict surface syntax trees. For shallow semantic tasks on the other hand, such as biomedical text mining, open domain relation extraction, or unsupervised semantic parsing, the relations between content words are more important than the overall tree structure. These tasks use collapsed or CCprocessed SD representations, which may be graphs instead of trees, and may contain additional and augmented relations. E.g. in “Fred started to laugh”, the relation between the controlled verb

laugh and its controller, *Fred* is made explicit in the CCprocessed SD representation.

The enhanced UD representation has the following features:

- additional relations and augmented relation names,
- Augmented modifiers. The collapsed SD graphs also include the preposition in the relation name. This helps to disambiguate the type of modifier. All nominal modifiers (nmod) also include the preposition in their names. The same is true for more complex PPs which are either analyzed as adverbial clause modifiers (advcl) or as adjectival clause modifiers (acl). Conjunct relations are augmented, e.g. conj:and.
- Propagated governors and dependents to clauses with conjoined phrases, and
- Subjects of controlled verbs linked.

2.4.8.1 *The enhanced++ UD representation*

The enhanced++ UD representation is more interesting for natural language understanding systems that try to extract relationships between entities, e.g those in open domain relation extraction, or relationships between objects in image descriptions.

Partitive noun phrases are phrases such as *both of the girls*, in which *both of the* acts semantically as a quantificational determiner. In the basic UD representation, however, *both* is the head while *both girls* is headed by *girls*. In order to obtain a similar analysis for both phrases, enhanced++ UD changes the structure of the basic dependency trees, which is not allowed according to the guidelines for enhanced dependency graphs. They treat the first part of the phrase as a quantificational determiner promote the semantically salient NP to be the head of the partitive and analyze the quantificational determiner as a flat multi-word expression that is headed by its first word the quantificational determiner is attached using the special relation det:qmod. *Light noun constructions* such as *a panel of experts* or *a bunch of people* are treated similarly.

Multiword prepositions such as *in front of* traditionally contain a relation between house and front, and front and hill. Here the enhancement++ lies in representing the relation between house and hill.

Conjoined preposition Such as *I bike to and from work* also pose some challenges. Ideally there is an nmod:to as well as an nmod:from relation: *bike to work* and *bike from work* are conjoined by *and* CCprocessed Stanford Dependencies representation introduced copy nodes which enhanced++ UD adapts 'I bike and bike 0 to and from work, respectively'. *Conjoined prepositional phrases* such as *She flew to Bali or to Turkey* should encode that the two nmod:to relations are conjoined

by *or*. For these reasons, enhanced++ UD also analyze such clauses with copy nodes.

Enhanced++ UD attaches both the referent of a *relative pronoun* directly to its governor, and the relative pronoun to its referent with a referent (ref) relation. E.g. the analysis of *The boy who lived* includes both '++boy ref lived' and '++boy nsubj lived'.

Enhanced++ UD does not propagate object or nominal modifier relations in clauses with conjoined verb phrases such as *the store buys and sells cameras* because of many cases such as *she was reading or watching a movie*. In contrast to AMR, enhanced++ UD does not distinguish between comitative and instrumental: AMR requires SRL which is very hard.

Enhanced++ UD is limited regarding generalized quantifiers and controlled verbs, such as *Everybody wants to buy a house* 'Everybody nsubj:xsubj buy' where the UD graph encodes approximately 'Everybody wants that everybody buys a house'. The graph for *Everybody sleeps or is awake* approximately encodes 'Everybody sleeps or everybody is awake'. Another limitation regard whether a conjoined subject (*Sue and Mary are carrying a piano*) should be interpreted distributively or collectively, which depends on world knowledge and the context.

2.4.9 *The State of the Art in Semantic Representation*

We conclude this chapter with recent overviews of semantic representation schemes. The first one, Abend and Rappoport (2017) clarify the general goals of research on semantic representation (except for vector space models), and compare them with syntactic schemes.

The paper discusses the goals of semantic representations (SRT), the components, (predicate-argument relations, discourse relations and logical structure), the concrete SRT schemes and annotated resources, the criteria for evaluation, and the relation to syntax. They focus on the level above the words, i.e. meaning relationships between lexical items, rather than the meaning of the lexical items themselves. The main differences between SRTs are the formalism and interface with syntax, the ability to abstract away from formal and syntactic variation, the level of training required for annotators, and the level of cross-linguistic generality.

In Abend and Rappoport's view, SRTs should be paired with a (computationally efficient) method for *extracting* information from them that can be directly evaluated by humans. *Applications* include inference, as in textual entailment or natural logic; supporting knowledge base querying; and defining semantics through a different modality, images, or embodied motor and perceptual schemas. (They defer sentiment.)

2.4.9.1 *Semantic Content*

As we have seen in Section 2.4.2, *events* (sometimes called frames, propositions or scenes) include the predicate (main relation, frame-evoking element), arguments (participants, core elements) and secondary relations (modifiers, non-core elements). There are ontologies and lexicons of *event types* (also a predicate lexicon), which categorizes semantically similar events evoked by different lexical items, e.g. FrameNet, which defines frames as schematized story fragments evoked by a set of conceptually similar predicates, or the Richer Event Descriptions framework. This notion of events should not be confused with events as defined in Information Extraction and event coreference, such as a political or financial event.

SRTs differ in which *nominal and adjectival predicates* are covered. Recent versions of PropBank covers eventive nouns and multi-argument adjectives. FrameNet covers all these, but also covers relational nouns that do not evoke an event, such as "president". SRTs may represent arguments that appear outside sentence boundaries, or do not explicitly appear anywhere in the text.

Core and non-core arguments are distinguished semantically rather than distributionally. Core arguments are whose meaning is predicate-specific and are necessary components of the described event, while non-core arguments are predicate-general. FrameNet defines core arguments as conceptually necessary components of a frame, that * make the frame unique and different from other frames; and peripheral arguments, which introduce additional, independent or distinct relations e.g. time, place, manner, means and degree.

Semantic roles in FrameNet are shared across predicates that evoke the same frame type, e.g. "leave" and "depart"; PropBank roles are verb-specific, and the set was extended by subsequent projects such as AMR; and VerbNet and subsequent projects use a closed set of abstract semantic roles for all predicate arguments, such as AGENT, PATIENT and INSTRUMENT.

Abend and Rappoport discuss *temporal relations* in details. This kind of analysis may mean timestamping according to time expressions found in the text, or by predicting their relative order in time. The main resources are TimeML, a specification language for temporal relations; and annotated corpora by the TempEval series of shared tasks. The theory goes back to scripts, schematic, temporally ordered sequences of events associated with a certain scenario, e.g. going to a restaurant Causal relations between events have applications (including planning and entailment) and annotation schemes, also integrated with TimeML-style temporal relations. The internal temporal structure of events has been less frequently tackled, but Moens and Steedman (1988) defined an ontology for the temporal components e.g. a preparatory process (e.g., "climbing a mountain") and its culmination ("reaching its top").

Statistical work on this topic is unfortunately scarce but involves aspectual classes, and tense distinctions.

Spatial Relations have their cognitive theories and applications in geographical information systems or robotic navigation. The task of Spatial Role Labeling with its shared task SpaceEval subsumes the identification and classification of places, paths, directions and motions, and their relative configuration.

In the papers running example, *Although Ann was leaving, she gave the present to John.* the leaving and the giving events are sometimes related through 'CONCESSION', evoked by "although". *Discourse* analysis is useful but overlooked for summarization, machine translation and information extraction. Resources include the Penn Discourse Treebank, which classifies the relations between discourse units using high-level relation types like TEMPORAL, COMPARISON and CONTINGENCY; and finer-grained ones such as JUSTIFICATION and EXCEPTION. This resource focuses on local discourse structure. RST Discourse Treebank puts more focus on higher-order discourse structures and deeper hierarchical structures. A narrower but better suited field is the segmentation of scientific papers into parts like background and discussion. Some schemes, e.g. GMB and UCCA, support cross-sentence semantic relations.

Logical structure, i.e. quantification, negation, coordination and their associated scope are important in applications that require mapping text into an executable language, such as * a querying language or robot instructions, and in recognizing entailment relations. Approaches to *inference and entailment* include Recognizing Textual Entailment, and Natural Logic with different annotation principles and resources.

2.4.9.2 *Semantic Schemes and Resources*

- Semantic Role Labeling.
- As we saw in Section 2.4.7, AMR has predicate-argument relations, including semantic roles (adapted from PropBank) that apply to a wide variety of predicates (including verbal, nominal and adjectival predicates), modifiers, co-reference, named entities and some time expressions, but currently no relations above the sentence level. It is English-centric, which results in an occasional conflation of semantic phenomena realized similarly in English, and difficulties with invariance across translations.
- Universal Conceptual Cognitive Annotation (Abend and Rappoport 2013) is a cross-linguistically applicable scheme for semantic annotation, building on typological theory, primarily on Basic Linguistic Theory. It includes argument structures of various types and relations across languages, but no semantic role information. UCCA distinguishes between primary and aspectual verbs e.g. *happen to*, and supports annotation by non-experts.

- Universal Decompositional Semantics provides semantic role annotation, word senses and aspectual classes (e.g., ±realis) collected through crowd-sourcing. UDS uses feature bundles e.g. +volition and +awareness, rather than agent.
- The Prague Dependency Treebank (PDT) Tectogrammatical Layer (PDT-TL) represents argument structure (including semantic roles), tense, ellipsis, topic/focus, co-reference, word sense disambiguation and local discourse information.
- CCG-based Schemes.
- HPSG-based Schemes use feature bundles. Annotated corpora and manually crafted grammars exist for multiple languages along with broad-coverage Semantic Dependency Parsing shared tasks and corpora.
- OntoNotes has multiple inter-linked layers of annotation, borrowed from different schemes.

Citations can be found in the original paper.

UNIVERSALITY Besides remarkable cross-lingual resources like BabelNet, UBY, and Open Multilingual Wordnet, SRL schemes and AMR have also been studied for their cross-linguistic applicability. PropBank and FrameNet have been translated to multiple languages, and there are SRT schemes that use cross-linguistic applicability as main criteria, e.g. UCCA, and the LinGO Grammar Matrix, both of which draw on typological theory.

2.4.9.3 *Anchoring graph fragments to tokens*

Finally, we would like to follow Koller, Oepen, and Sun (2019) in distinguishing three flavors by the degree of anchoring. The strongest form of anchoring is bi-lexical dependency graphs, when graph nodes injectively correspond to surface lexical units (tokens). In such graphs, each node is directly linked to a specific token (but there may be semantically empty tokens), and the nodes inherit the linear order of their corresponding tokens. Linguistic frameworks in this flavor include CCG word–word dependencies, Enju Predicate–Argument Structures, DELPH-IN MRS Bi-Lexical Dependencies, and Prague Semantic Dependencies.

The middle flavor relaxes the correspondence relations between nodes and tokens, while still explicitly annotates the correspondence between nodes and parts of the sentence, but nodes may align with subtoken or multi-token sequences, e.g. (derivational) affixes or phrasal constructions. Nodes may correspond to overlapping spans, enabling lexical decomposition (e.g. of causatives or comparatives). Representatives include Universal Conceptual Cognitive Annotation and two variants of 'reducing' the underspecified logical forms into directed graphs.

AMRs on the other end are unanchored, in that the correspondence is not explicitly annotated. AMR deliberately backgrounds notions of compositionality and derivation. The framework frequently invokes lexical decomposition and represents some implicitly expressed elements of meaning, abstracting furthest from the surface signal.

Mi generáltunk. Legalábbis azt hittük, hogy generálunk.
'We generated. At least we thought we generated.'

— Ferenc Kiefer on generative linguistics before Chomsky (1970).

3

THE 4LANG SEMANTIC NETWORK

Contents

This chapter introduces the `4lang` semantic network, which represents linguistic meaning (words and greater linguistic units) in the form of graphs. `4lang` has been developed in the Human Language Technologies Research Group Budapest since 2010. The name refers to that the core dictionary, the main topic of this chapter, has bindings in four languages, representatives "of the major language families spoken in Europe; Germanic (English), Slavic (Polish), Romance (Latin), and Finno-Ugric (Hungarian)[1] [...] `4lang` is an algebraic (symbolic) system that puts the emphasis on lexical definitions at the word and sub-word level, and on valency (slot-filling) on the phrase and sentence level" (Recski et al. 2016). These two levels are discussed in this chapter and Chapter 6 respectively.

"Historically, `4lang` falls in the AI/KR tradition, following on the work of Quillian (1969, Section 2.1.1), Schank (1975, Section 2.1.3), and more recently Banarescu (2013, Section 2.4.7). Linguistically, it is closest to Wierzbicka and Goddard (1972, 2002, Section 2.2.3) and to modern theories of case grammar and linking theory (see Butt (2006) for a summary)." (References to sections in the present thesis added.)

1 Kornai (2022) adds Japanese and Chinese. "The relative ease of creating these new bindings goes some way onward ameliorating concerns of eurocentricity."

While this chapter belongs to the background part of the thesis, it also reports work done by the author: In the first version of `4lang`[2], the manually written definitions were mostly created by Makrai (first described for the Hungarian NLP community in Kornai and Makrai (2013)). The characterization of the importance of each concept in the recursive process of word definition in Section 3.3 originally appeared as Makrai (2013).

3.1 NODES AND EDGES

3.1.1 *Concepts*

The backbone of `4lang` consists of 1942 defined words and bound morphemes from the Longman Defining Vocabulary. The version of the Longman dictionary that was available to us (Bullon 2003) uses other elements, so we have further expanded the vocabulary with 197 simple words (e.g. *dimension, two, communicate, conform , mammal, item, artefact*), with 188 proper names, the definition of which is essentially just a reference to the corresponding element of the encyclopedia (e.g. *Greenland, Greenwich, Guy Fawkes*) and 147 compounds (*bell-shaped, bitter-tasting, blue-black*). The latter are uninteresting from our present perspective.

The definitions in `4lang` were made by human labor, mostly using classical dictionaries as well, most notably the Longman dictionary. Probably the biggest novelty in writing definitions is radical *monosemy*. This means that `4lang` tries to grasp the abstract meaning of the words, from which specific uses can be deduced. In Kornai and Makrai (2013) we cited the definition of *potash* from Webster's Third (Gove 1961) to show how words considered to be polysemous are defined in traditional lexicology. *Potash* has four meanings there. At the Doctoral Conference on Applied Linguistics (Makrai 2013), we provided a similar example from the English WordNet (Miller 1995) with six meanings of the word *stomach*.

Nodes in `4lang` graphs represent concepts that are more abstract that usual lexical items in three aspects: they are language-independent, monosemic and free of syntactic type.

According to the principles of `4lang`, most words are monosemic. Disambiguation is only done for pure homonyms, e.g. the word form *state* corresponds to separate entries in the senses related to 'country' and 'status'. It follows from momosemy that, since `4lang` describes the conceptual meaning, `4lang` contains a single concept where two words differ only in their part of speech, e.g. action nouns are the same concept as the verbal stem. This approach obviously deviates from Montague grammar, where syntactic types correspond to semantic types.

2 https://github.com/kornai/4lang/tree/1d19f167b9c0eace5bd874759860781be78f96ed

The 4lang dictionary strives to be *language-independent*. When defining the words, we tried to take into account a couple of languages, and the word forms of the terms were indicated in Hungarian, English, Latin and Polish. Colleagues have expanded the dictionary to forty languages (Ács, Pajkossy, and Kornai 2013).

Language-independence may be contrasted with the Saussurean definition of a linguistic sign which is an ordered pair consisting of a cluster of (spoken or written) forms in a specific language and an extra-linguistic category in the mind. Whether human categorization is dependent of the mother tongue and other languages learned by the speaker early on is a classical topic in psycho-linguistics. The engineer would observe that people can express the same content in any language, and the greatest problem one faces in finding translational equivalents is that an ambiguous word in some language may (not surprisingly) translate to some other language in multiple ways, depending on context.

Thus language-independence is related to 4lang's approach to ambiguity (polysemy). Rather than including in disambiguation as much information on different uses as possible, we prefer representing each surface morpheme with a single graph, a method called *monosemy* in the programmatic book by Ruhl (1989). Elements of the meaning of a word in a context that is not present in the abstract lexical item should be deduced from the similarly abstract representations of context words. We think that computing the meaning of usages of words that are usually called metaphoric is the basic mechanism behind human linguistic capabilities, and artificial understanding should work with a similar goal, including computational models of extra-linguistic knowledge and pragmatic implicatures apart from the lexicon. How the distinction between polysemy and homonymy can be made on the basis of data and word embeddings will be discussed in Chapter 8 in the frame of multi-lingual word sense induction, the computational task of clustering word occurrences to lexical items based on multi-lingual corpora.

3.1.2 *Syntactic and semantic type*

4lang being a conceptual network implies that representations try to factor out pure morpho-syntactic differences on the word level.[3] This avoidance of types is in contrast to lexicographic practice, both traditional or symbolic computational, that splits usages of words by parts of speech.

The 4lang approach to the lexicon is illustrated by the phenomenon that a great part of the English core vocabulary consists of words that

3 The interested reader may learn about the syntactic part of the 4lang theory, motivated by functional programming and formalized in Eilenberg Machines, in Section 6.3.2 of Kornai (2008) and in Kornai (2019).

appear as nouns and verbs as well, with semantically equivalent meanings (*divorce*$_N$ is exactly the situations where some people *divorce*$_V$). The corresponding pairs in Hungarian are derivational ones: remaining with the same example, the noun *vál-ás* is derived from the verb *vál(ik)* by a compositional suffix.

Formal semantics is organized along the principle of compositionality: the representation of a phrase or a sentence needs to be computed from the representations of the immediate constituents and the way of their composition. Montague Grammar formalizes the compositional requirement by associating rewrite rules over syntactic forms to semantic rules. Terminals of the semantic sub-grammar are semantic types, most notably entities and truth-valuable states of affairs.

Compositionality also applies to `4lang` graphs. Representations of phrases and sentences are composed of those of the words, and formulas in the hand-written core vocabulary we discuss in Section 3.2 are parsed to graphs in a rule-to-rule fashion. The main operation in both is to draw a link from a node in the graph corresponding to the macro-structure of the linguistic unit to the so-called head-node of the constituent.[4]

The lack of semantic types can be seen as radical lexicalism: `4lang` concentrates on the meaning of words and phrases at the expense of type consistency in the graph. The noun-verb *cook* for example means that 'a person makes some food'. This definition is less exact that those applying the POS distinction, e.g. the verb apparently implies that heat is used while the noun does not. A greater problem is that the head-node depends on the POS: it has to be 'person' if the noun is meant, and 'make' if the verb. We insist on using a single representation. We can choose one of the head-node candidates arbitrarily (practically the one corresponding to the most frequent POS of the word), and admit that graphs that we build from sentences with the other POS(s) will be somewhat inconsistent. Whether the resulting representations still capture enough lexical content to be useful in application has to be tested in empirical fashion.

3.1.3 *Edges*

In the `4lang` meaning representation framework, the meaning of words and greater linguistic units is formalized in pointed directed graphs with nodes labeled by concepts and edges colored in three colors 0, 1, and 2. Pointedness means that one node, the *head*, is distinguished for compositional purposes, as already discussed in 3.1.2.

4 The theory may allow the link to point to a sub-graph, motivated by *accusativus cum infinitivo* sentences like *I see the father coming* where the object of seeing can argued to be the seeing of the father as well as the father itself, but this idea is not implemented and not part of this thesis.

Woods (1975) argues that a too large inventory of edge types (colors) makes reasoning with graphs computationally unfeasable. This problem is avoided in `4lang` by splitting relations to various levels. At the deepest level, there are only three types of edges (0, 1, and 2). When there is an edge $c_1 \xrightarrow{i} c_2$ with label $i \in \{0, 1, 2\}$ from concept c_1 to concept c_2, we will also say that c_2 is on the ith *partition*[5] of c_1. Binary relations, which are the topic of Chapter 5, are represented with nodes (typically with 1 and 2-edges leading out of them to their first and second argument). Ternary and higher arity relations are decomposed (Kornai 2012) to at most binary ones with methods pioneered in generative semantics, e.g. 'give' is represented as 'cause to have'. Finally, deep cases, typically verbal roles (the topic of Chapter 6) are also represented by nodes with special labels, e.g. an `=AGT` node in the graph representing a verb is a place-holder for the representation of the verb's agent.

Turning to the edge-colors, 0 denotes every relation in which a concept modifies some other as a whole: we draw an abstraction over the traditional genus/hypernym/IS_A (e.g. `dog` $\xrightarrow{0}$ `animal`), (generic) unary predication (`dog` $\xrightarrow{0}$ `bark`), and attribution (`dog` $\xrightarrow{0}$ `faithful`). The interested reader may learn more about `is_a`, genus, and hypernym in Section 4.5 of Kornai (2019). 0 is used for verbs as well as nouns. Unlike Levelt, Roelofs, and Meyer (1999), where `escort IS-TO accompany`, in `4lang` we simply state that `escort` $\xrightarrow{0}$ `accompany`.

1 and 2 represent two arguments of a function that play asymmetric roles, e.g. the agent and patient role of a verb (e.g. `cow` $\xleftarrow{1}$ `make` $\xrightarrow{2}$ `milk`), or the figure and the ground in tempo-spatial relations (`star` $\xleftarrow{1}$ `at` $\xrightarrow{2}$ `sky`).[6] Nodes (concepts) with an 1 or 2-labeled out-edge will be called *binary*, while the rest will be called *unary* because these concepts correspond to unary predicates of truth-conditional logic.

We note that in creating the definitions, we sought to record only linguistic knowledge, so to speak, *analytical* truths. The 4lang definition of *night* states that there is no sun. By this we also say that if the sun come up at *night* (say at midnight), we would no longer call that period *night*.

5 Those who are familiar with gold-age meaning representation, especially Hendrix (1975), should note that in `4lang`, partition is meant much more simply that for Hendrix, who introduced a machinery with the same name to provide an adequate quantification mechanism for semantic network concepts. In `4lang`, more concepts on a partition of a concept (out-neighbours with a fixed edge label) are interpreted as a conjunctive boundle of properties.

6 It can be argued that in terms of predication, the direction of the 0 versus 1 and 2 edges is somewhat inconsistent: in `dog` $\xrightarrow{0}$ `animal`, the link goes from the argument to the predicate, while in `cow` $\xleftarrow{1}$ `make` $\xrightarrow{2}$ `milk`, the edges lead from the function to the arguments. In the view of the thesis author, this discrepancy may be an accident in the development of the system, but need not corrupt empirical results in applications. Nevertheless, András Kornai writes in personal communication that what the argument and what the predicate is in the case of 0, and also in the case of intransives in general, is debatable/changeable, e.g. in the first two articles of Montague, there is boy(sleep) and sleep(boy), respectively. It can be argued for both.

3.2 THE DEFINING VOCABULARY

Symbolic representations define concepts by other concepts. Some methods take this circularity as a basic property of language, while others break it by using primitives, words that play the same role in semantics as primitive notions do in mathematics. The first approach includes disciplines ranging from structuralist semantics to semantic networks (Chapter 2) and information retrieval (Section 3.3). The primitive-based approach is exemplified in this thesis by the Natural Semantic Metalanguage (Section 2.2.3), and the Longman Defining Vocabulary (Section 2.3.4). The `4lang` approach is closer to the latter, but it is important that we do not specify the defining vocabulary on theoretical grounds, but we derive it from the definition graph (Section 2.1.6) with an iterative process (see András Kornai et al. (2015, Section 2.1), Ács, Nemeskey, and Recski (2017, Section 2.2), and Kornai and Makrai (2013), the latter is in Hungarian).

The meaning of a sentence is composed from that of the words in it, but the word inventory is still too great to give a `4lang` account of each item. Now we describe our method for vocabulary reduction from the, say, 80–160 thousand (disambiguated) words in a traditional dictionary to a defining vocabulary for which we create `4lang` representations manually, constituting the main contribution in this chapter.

It must be noted that members of the defining vocabulary are not primitives of definition. This is in accordance with more approaches: the structuralist notion of word sense; that "the full meaning of any concept is the whole network as entered from the concept node" (Collins and Loftus 1975); and what Lenat and Guha (1990) say about the lack of primitive actions in Cyc: "actions are not merely macros introduced for notational convenience, for use instead of more complex sequences of primitive actions. [Our] approach is motivated by two reasons: we wish to be able to reason at different levels of abstraction and a priori assigning of a set of actions as primitives goes against this".

Our methods for defining the whole vocabulay in terms of a more restricted set (as well as previous work in this field) are discussed in Section 2.1 of András Kornai et al. (2015). There are two basic approaches: bottom-up methods use a defining vocabulary specified on some theoretical basis, but our group has done top-down computations as well to discover the defining vocabulary of both traditional dictionaries and our manually written definitions themselves.

> The first modern efforts in [the direction of a basic vocabulary] are Thorndike (1921)'s Word Book, based entirely on frequency counts (combining TF and DF measures), and Ogden (1944)'s Basic English, based primarily on considerations of definability. The Swadesh (1950) list puts special emphasis on cross-linguistic definability, as its primary goal is to support glottochronological studies.

> The idea that there is a small set of conceptual primitives for building semantic representations has a long history both in linguistics and AI as well as in language teaching. The more theory-oriented systems, such as Conceptual Dependency (Schank 1972) and NSM (Wierzbicka 1985) assume only a few dozen primitives, but have a disquieting tendency to add new elements as time goes by (Andrews 2015). In contrast, the systems intended for teaching and communication, such as Basic English (Ogden 1944) start with at least a thousand primitives, and assume that these need to be further supplemented by technical terms from various domains. [...] A trivial lower bound [on the number of primitives] is given by the current size of the NSM inventory, 65 (Andrews 2015), but as long as we don't have the complete lexicon of at least one language defined in NSM terms the reductivity of the system remains in doubt.
>
> For English, a Germanic language, the first provably reductive system is the Longman Defining Vocabulary (LDV), some 2,200 items, which provide a sufficient basis for defining all entries in LDOCE (using English syntax in the definitions).

The core vocabulary of the `4lang` meaning representations framework is a set of about three thousand concepts with English, Hungarian, Latin and Polish exponents[7] and formal definitions that can be compiled to `4lang` graphs with the pymachine software package. The original vocabulary (words with ID up to 2692) was specified in the Hungarian Unified Ontology (MEO) Project based on theoretical considerations similar to those mentioned in the previous citation. This process is also described in the paper:

> We built a seed list composed of the Longman Defining Vocabulary (2,200 entries), the most frequent 2,000 words according to the Google unigram count (Brants and Franz 2006) and the British National Corpus, as well as the most frequent 2,000 words from Polish (Halácsy et al. 2004) and Hungarian (András Kornai et al. 2006). [For Latin,] we added the classic Diederich (1939) list and Whitney (1885)'s *Roots*.

Turning to the top-down method, in the same András Kornai et al. (2015), we formalized the defining vocabulary in graph-theoretic terms, based on the definition graph, whose nodes correspond to (disambiguated) words, and a directed edge $u \to v$ represents if v is used

7 Ács, Pajkossy, and Kornai (2013) describe how bindings in other languages can be created automatically.

in the definition of u. The mathematical formulation of the defining vocabulary is a feedback vertex set (FVS) that contains all nodes without out-edges (these are definitional primitives) and one node from each directed cycle. We found that in definition graphs there are much smaller FVSs that there may be if the graph was random: "For example, in the English Wiktionary, 369,281 definitions can be reduced to a core set of 2,504 defining words, and in Collins English Dictionary we can find a defining set of 6,490 words." Gold-age versions of the Longman Dictionary were created with a pre-specified defining vocabulary (LDV), what still shows its advantages in the newer, non-LDV-based version we have access to, as the defining vocabulary consists only of 1,061 words.[8] The interested reader may read more details on the possible gains of a smarter parsing of implicit cross references in dictionaries, handling compositional derivations of latinate stems, disambiguation, and multiword expressions in the paper. The key point is that a cca. 3000-word vocabulary that we defined with `4lang` formulas in the middle of the past decade. (Section 3.5) covers the defining vocabulary of traditional dictionaries. Further refinements of the `4lang` defining vocabulary can be found in Appendix 4.8 of Kornai (2019) and in Kornai (2022).

3.3 IMPORTANCE OF CONCEPTS IN THE DEFINITION GRAPH

3.3.1 *Introduction*

In this section, which originally appeared in Hungarian as Makrai (2013), we examine how important each concept is in sentence comprehension. We do this by transforming definitions that represent the meaning of words into a directed graph with concepts as nodes. Applying the method known in computer science as PageRank for the definition graph described in this article, the values assigned to each vertex can be interpreted as the importance of the corresponding concept in understanding other words and phrases. The PageRank method was originally introduced to measure the relevance of websites. The structure of the article is as follows. In Section 3.3.2 we present the definition graph, and in Section 3.3.3 we present the method PageRank used to calculate the weight of each concept. Finally, we report the numerical results in Section 3.3.4.

We work at the word level, yet it is important to talk about arguments of words (typically verbs and relational nouns). As it follows from the principle of compositionality, we require that the representation of the meaning of a structure consisting of a function and its arguments is composed from the representation of the meaning of the function and the arguments. To ensure this, the definition of functions should indicate where the representation of each argument has to be inserted. We

8 This set roughly corresponds to the words that are marked with `u` in the 6th column of the `4lang` file.

do this by referring to the deep cases of the arguments. (Fillmore 1968). In this chapter, the names of deep cases abbreviate Hungarian surface cases, e.g. NOM denotes the subject, ACC denotes the object, DAT denotes the dative argument, and OBL denotes the oblique. Chapter 6 develops a more theoretically grounded system (more easily comparable to Fillmore's idea and also grasping alternations). Recall that 4lang accounts for the meaning of ditransitive (and higher arity) verbs in the syntax using predicates of at most two variables. See Kornai (2012) for details.

3.3.2 *The definition graph*

In this section, we first show how we transformed the 4lang dictionary into a directed matrix, which enabled us to characterize the semantic importance of the concepts. Thereafter we describe the graph.

The vertices of a definition graph are concepts in the dictionary, and if, for example, the word 'metal' is used in the definition of 'steel', then a directed edge in the graph points from the vertex corresponding to the latter to that corresponding to the former. The graph has 2,897 vertices and 7,816 edges (i.e. there are relatively few edges between two vertices).

The mathematical concept of strongly connected components will play an important role later. Two vertices are called *strongly connected* if a path (a sequence of edges) leads from them to each other. This relation is an equivalence relation, it classifies the vertices into classes, which are called *strongly connected components*. The strongly connected components of the 4lang graph are interesting in themselves as they give an intuition about the graph, so we briefly present them.

# nodes	#	
662	1	{yellow, four, sleep, under, lack, month. . . }
12	1	{January, February, . . . , December}
7	1	{Monday, Tuesday, . . . , Sunday}
5	1	{furniture, chair, table, bed, cupboard}
4	3	{queen, royal, monarch, king}, {cereal, flour,. . . },. . .
3	8	{male, sex, female}, {calm, disturb, upset},. . .
2	26	{exist, real}, {reason, cause}, {child, parent},. . .
1	2302	{PART_OF}, {other}, {IS_A}, {number}, . . .

As Section 3.3.2 shows, the largest strongly connected component consists of quite mixed words (*yellow, four, sleep, under, lack, month dots*). The next largest strongly connected components are given by cycles such as months or days of the week. The definition of e.g. months consists of the information that they are months and the previous and the next month. The next greatest strongly connected components are related to a concept, e.g. kinds of furniture and the word *furniture* itself

form a strongly connected component, as the word furniture is included in the definition of kinds of furniture and some kinds of furniture are provided in in 'furniture' as examples. Finally, most concepts have no out-edges (i.e. they are primitives of definition).

3.3.3 *Weighting the concepts*

Circularity is an old problem of lexicography: if we say that a 'child' is one who has a 'parent' and 'parent' is the one who has a 'child', we have not said much. Modern dictionaries avoid this problem by limiting the vocabulary of the definitions to the so called *defining vocabulary*. We choose the opposite direction by characterizing the importance of defining words based on the dictionary, as they define each other.

The mathematical method used for this can be thought about as a random walk in the definition graph. We start the walk in a randomly chosen concept (the probability distribution by this start concept is chosen, as we will see, does not matter). During the steps of the walk, we randomly take one of the concepts defining the current concept with an even distribution (more precisely taking into account the multiplicity). By the limit distribution of a random walk we mean the probability that we will be in each concept (node) after a long time. This just expresses how important a given concept is in defining all the concepts, taking into account recursively the importance of the concepts to be defined.

The limit distribution is unique (that is, independent of the initial distribution) if and only if the graph consists of a single strongly connected component. We have seen that this is not the case in the `4lang` graph. PageRank is for weighting of the vertices of graphs consisting of more strongly connected components. Intuitively, during the walk behind PageRank, with probability less than 1 you still go to one of the vertices directly accessible from it (random, including multiplicity), but you can go to any of the peaks. For transition probabilities, this means that if we go to node j with probability $P(i, j)$ in the original walk, given we are currently in the vertex i, the same transition probability in the new walk will be

$$P_d(i, j) = \frac{1 - d}{n} + dP(i, j)$$

where d is the so-called damping factor (most often $d = 0.85$) and n is the number of nodes. As d goes to 1, the limit distribution approximates that of the original matrix.

3.3.4 *Results*

Of course, the PageRank value depends on the damping factor d. We first state results that are independent of different values of d, and then we present the differences.

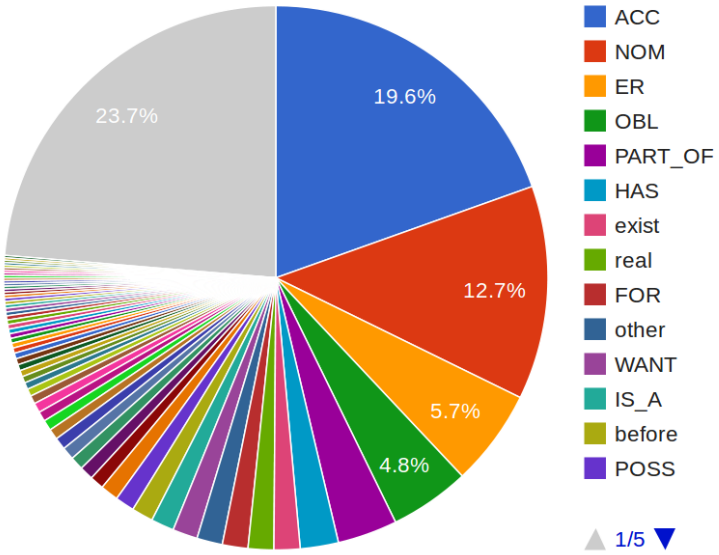

Figure 5: PageRank of concepts. Concepts are sorted by PageRank. The triangles with "1/5" in between below the legend can be disregarded.

Figures 5 and 6 show the results. Some (few) items were given quite a lot of weight and very many got very little. The figure on the right shows the Page Rank of the 17 most important concepts on a logarithmic scale. Here, the horizontal axis is ranked by importance ($d = .1$), and the vertical logarithmic axis is the PageRank value of the corresponding ranked element. The two most important ones are two deep cases, in line with the intuitive idea that understanding the arguments of a verb structure plays a significant role in understanding the structure. We would like to highlight two more bivariate predicates: `PART_OF`, which may be familiar to the reader from the meaning representation literature, and the `FOR` expressing the goal (the purpose of things related to man). These results promise that in order for artificial intelligence to be able to draw the right conclusions, it must first handle the items at the top of the rankings well.

The PageRank of the 16 most important elements does not depend much on the damping factor. The only exception is possession (`HAS`), which appears in many definitions (19% of the definitions are involved), but it has no definition itself. Surprisingly, `HAS` gets a high PageRank for low damping (for example $d = 0.85$) and low for strong damping.

Characterization of defining vocabulary is an old problem in lexicology. Kornai and Makrai (2013) sets the definition of the basic vo-

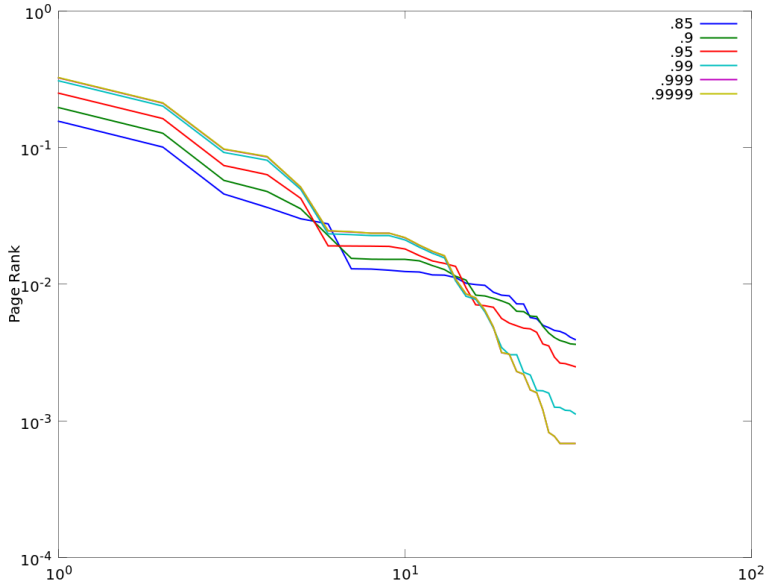

Figure 6: PageRank of concepts. Concepts are sorted by PageRank. If the nth concept has PageRank p, it is shown at $x = \log n$, $y = \log p$. Colors correspond to different damping factors.

cabulary as one of the main aims of `4lang`. This section can thus be summarized as the `4lang` contribution to the numerical characterization of the defining vocabulary, i.e. a formal definition of this much discussed set.

3.4 A MODEL OF NAIVE REALITY

What kind of information is included in `4lang` representations? Language philosophy and lexicography distinguish word meaning from other kinds of knowledge, while cognitive science and NLP put the emphasis on grounding linguistic knowledge in other capabilites of natural or artificial intelligence such as vision and memory.

Kant (1781) introduced the distinction between analytic propositions that are true by virtue of their meaning (*All bodies occupy space.*), and synthetic propositions that are true of their references in the real world (*All creatures with hearts have kidneys.*). Within synthetic propositions, *a priori* and *a posteriori* propositions can be distinguished based on whether their justification relies upon experience. Logical positivists revisited the definition of analytic proposition as a proposition that is made true (or false) solely by the conventions of language.

W. v. Quine (1951) argued that the analytic–synthetic distinction is untenable despite "one [being] tempted to suppose in general that the truth of a statement is somehow analyzable into a linguistic component and a factual component." Wikipedia summarizes Quine's ar-

gument so that the notion of an analytic proposition requires a notion of synonymy (e.g. the proposition 'Bachelors are unmarried' is analytic because *bachelor* is synonymous with something like *older unmarried man*), but establishing synonymy inevitably leads to matters of fact via semantic equivalence.

Grice and Strawson (1956) offer a pair of thought experiments to restore the distinction. The protagoinst of the first experiment says that *My neighbor's three-year-old child understands Russell's Theory of Types.* The other one say *My neighbor's three-year-old child is an adult.* The intended distinction is that it is *logically* impossible for a child of three to be an adult, and its *naturally* impossible for a child of three to understand Russell's Theory of Types. "In both cases we would tend to begin by supposing that the other speaker was using words in a figurative or unusual or restricted way; but in the face of [their] repeated claim to be speaking literally, it would be appropriate in the first case to say that we did not *believe* [them], while in the second case [we would] say that we did not *understand* [them]."

For a deeper understanding of the `4lang` principle that the lexicographer should record analytic properties and disregard synthetic ones, the reader may refer Section 5.7 of Kornai (2019), which heavily builds on the philosophical work in Putnam (1976), who "restored the honor of the analytical/synthetic distinction".

Our system is similar to truth-conditional semantics in that it needs some model. More exactly, there are more models, an internal one modeling linguistic meaning, and external models in charge of specific in-domain knowledge and reasoning. The internal model is different from that of modern sciences. E.g. the `4lang` definition of *heart* includes, besides the scientific truth that 'heart is an organ' and 'heart moves blood' the naive fact that 'love is in heart', or we define *death* as the end of life, though theology may state that life continues after death. As a third example, 'speed' is related to 'move' in `4lang`, but the exact nature of this relation which is explained in physics is not part of the naive world model neither can be expressed in `4lang`.

Gruber et al. (1993) defines an ontology as a formal, explicit specification of a shared conceptualization. In such Knowledge Representational terms, the core definitions that are the central topic of this chapter constitute the top-level ontology of the `4lang` meaning representation framework, keeping in mind that at this top level, we concentrate on linguistic meaning, and domain-specific knowledge can represented in external models.

3.5 FORMULAS

The main contribution of this chapter is the representation of a cca. 3000-word core vocabulary that, according to computations discussed in Section 3.2, is sufficient to define all the words in a dictionary. These

core representation are written in `4lang` formulas that are compiled to `4lang` graphs by the pymachine software package.

`4lang` representations are graphs whose nodes are labeled by (mainly alphabetical) strings, the exponents of the concept that the node represents, edges have one of the colors 0, 1, and 2, and one nodes is distinguished as a head-node. Such graphs can be specified by listing the nodes and the edges, but we maintain a formula representation as well which is more reminiscent of natural language definitions found in a dictionary. In this section, we describe the syntax of these formulas we call a *minisyntax* along with the graphs they are compiled to in `pymachine` (*minisemantics*). The minisyntax and the minisemantics together will be called *minigrammar*. (The terminology *metasyntax*, *metasemantics*, and *metagrammar* may be more familiar as they are the syntax and the semantics of some metalanguage, the object languages being natural languages, but we think that *meta* would suggest something impressive while minigrammar is a modest mechanism for creating `4lang` graphs in lexicographer-friendly fashion.)

The minigrammar was first published in András Kornai et al. (2015) with the shortcoming that we did not make the head-node explicit, which made the formalism somewhat unclear. In Figure 7 we reproduce the grammar published there with some simplification in the system of non-terminals and indicating the head-node in each graph. The left column specifies how the graph representing the definiendum is built. There is always a *definiendu**m*** node denoted with m (labeled by the definiendum). The right column shows how a graph $g(X)$ representing the non-terminal X in the left side of the corresponding rule can be build from the graphs $g(Y)$ representing the yields of the non-terminals Y in the right side of the rule and m by drawing the edges from the head-node of some $g(Y_1)$ or m to that of some $g(Y_2)$ or m. The head-node of the resulting graph is denoted by **boldface**.

Non-terminals of the minisyntax are D for definition, E for expression (subjunctive clause), E_u for "unary expression" (subjunctive clause with unary head), U for (lables of) unary nodes, B for (lables of) binary nodes, and A for arguments of binary nodes. The terminal `,` separates subjunctive clauses: a definition consist one or more clauses. Note that normal-font round parenthesises in this figure are used in regular expressions describing sentential forms, e.g. (`,`E)* is the Kleene-closure of `,`E, while the typewriter-font parens `(` and `)` are terminals of the minisyntax for 0-edges, e.g. `long(time)` compiles to `time` $\xrightarrow{0}$ `long`. Square brackets parenthize arguments of nodes, both those of unary nodes (`air[move]`) and those of binary ones (`actor IN/2758 [<theater>,<film>]`.

Most unary predicates are lower-case strings that may include _ – see below for special cases. Ambiguous word-forms are disambiguated by appending the terminal `/` plus a numerical id to the end, e.g. `light/739` is the opposite of dark(ness) while `light/1381` is the opposite of heavy. From the point of view of minigrammar, deep cases, the placeholders of

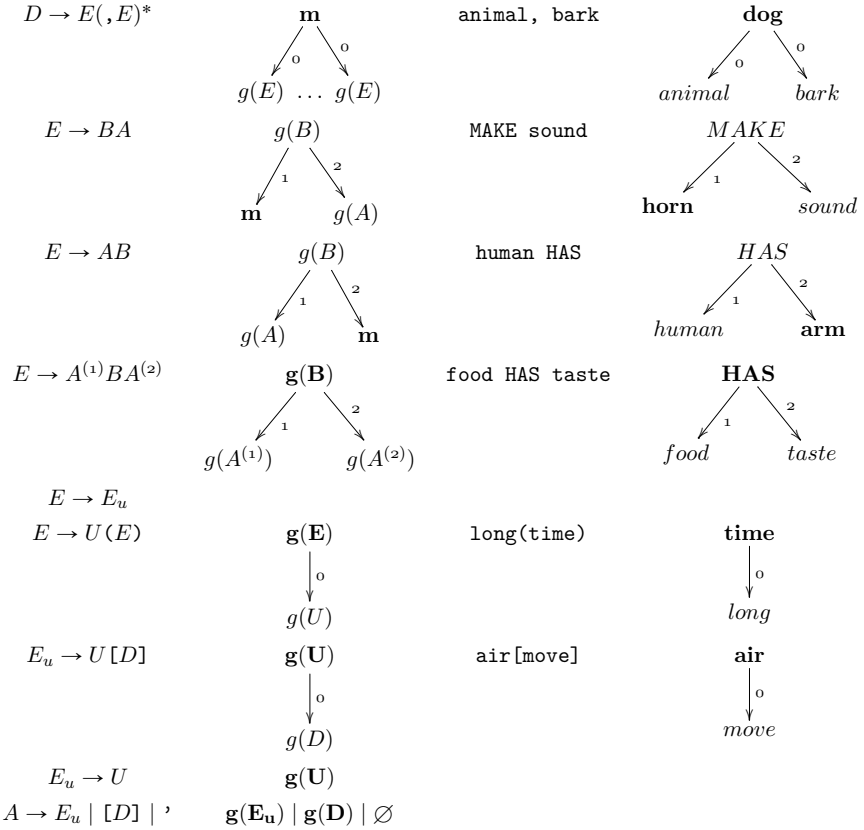

Figure 7: The original minigrammar

arguments in representations of functions, are also unary labels despite their linking purpose. Deep cases are type-set as e.g. `=AGT` or `=TO`. Some unary nodes are encyclopedic references, these are prefixed with the terminal `@`, e.g. `@United_States`. Binary node labels are uppercase string (type-set in this thesis in SMALL_CAPS for aesthetic purposes) also allowing `_`, e.g. HAS, PART_OF.

The first row corresponds to the top level: the definition of a concept is a conjunction of properties. For the theoretic background, see section 3.3. of Kornai (2019) who defines 'dog' as 'four-legged, animal, hairy, barks, bites, faithful, and inferior'[9]. The next three lines represent binary predication. By default, the definition parser in `pymachine` draws a 0-edge from empty arguments of a binary nodes to the definiendum m. This can be avoided by inserting the dummy argument `'` ($g(\text{'}) = \varnothing$). Nodes with the same label get unified unless there is the key-word `other` on their 0-th partition.

3.6 NEGATION

No summer's high
No warm July
No harvest moon to light one tender August night
No autumn breeze
No falling leaves
Not even time for birds to fly to southern skies

— Stevie Wonder

We conclude this chapter with a remark on the representation of (word-level) negation, which is related to inference. Woods (1975) specified the three main requirements for a semantic representation as the logical adequacy of the representation, the possibility of making inferences and deductions from the representation, and the ability of translating natural language text into the representation. A range of tasks have been proposed in the area of learning and applying commonsense knowledge. Inference with `4lang` representations have been tested in application by Recski et al. (2016) who use `4lang` graphs for measuring (semantic) similarity of words. (They discuss features in their Section 3.2.) From the configuration `train` $\xleftarrow{0}$ `vehicle` $\xrightarrow{0}$ `car` they infer that *train* and *car* are somewhat similar and from `park` $\xleftarrow{1}$ `IN` $\xrightarrow{2}$ `town` and `street` $\xleftarrow{1}$ `IN` $\xrightarrow{2}$ `town` that so are *park* and *street*. A key point in inference is inheritance: If we have `HAS wing` for all birds, `HAS wing` will also be true of all concepts for which $\xrightarrow{0}$ `bird` holds. Inheritance is closely connected to negation. Negation is expressed in `4lang` formulas by connecting a `lack` node to the oth partition of the property which

9 We give pre-theoretic meanings in 'single quotes', while `typewriter font` is kept for `4lang` formulas.

is lacking, e.g. by stating `diamond` $\overset{1}{\leftarrow}$ `HAS` $\overset{2}{\rightarrow}$ `color` $\overset{0}{\rightarrow}$ `LACK` in the definition of *diamond*, we escape the (contra-factual) inference of concluding `diamond` $\overset{1}{\leftarrow}$ `HAS` $\overset{2}{\rightarrow}$ `color` from the disjunction of `diamond` $\overset{0}{\rightarrow}$ `mineral` $\overset{0}{\rightarrow}$ `substance` and `substance` $\overset{1}{\leftarrow}$ `HAS` $\overset{2}{\rightarrow}$ `color`. A broader discussion of negation in `4lang` can be found in Chapter 4 of Kornai (2022).

Radim: "Was a story like nobody believed that it actually works, and you can do this sort of algebra with the vectors directly?"
Tomáš: "Oh, algebra, yeah."

— From a podcast with Tomáš Mikolov by Radim Řehůřek (21:20)

4

DISTRIBUTION AND VECTORS

Contents

Most contributions of this thesis are based on vector space language models (VSMs). In this chapter, we introduce VSMs as two interrelated families of word representations. The traditional method (Section 4.1.3) takes the co-occurrence matrix as a starting point, while more recent representations are learned as weights in shallow (Section 4.2) or deep (Section 4.3) neural networks. While this chapter belongs to the background part of the thesis, it also reports work with the author's contribution: Our Section 4.2.11 on sub-word embeddins for rich morphology originally appeared as Döbrössy, Makrai, Tarján, and Szaszák (2019).

The primary source of information about the meaning of a word is how often it is used in different contexts, an idea called the *distributional hypothesis* by linguists going back to Z. Harris (1951), and often quoted in the form that "You shall know a word by the company it keeps" (Firth 1957). The Saussurean definition of syntactic category (part of speech) is strikingly similar, the only difference in NLP practice appears to be how the context is defined (Sahlgren (2006), see Section 4.1.4): syntax is based on a short directed window (e.g. adjectives closely precede nouns) while semantic relations can be extracted from longer but symmetric windows (*dog* and *faithful* co-occur in sentences in any order).

One simple formalization of word distribution in a corpus is the *co-occurrence* matrix whose rows correspond to words in the vocabulary, columns to contexts, and cells contain the occurrence count of the word corresponding to the row appearing in the context corresponding to the column. What is meant by context depends on the application. In Latent Semantic Analysis (LSA, Deerwester, Dumais, and Harshman (1990), Section 4.1.3), columns of the original (unreduced) matrix correspond to documents. In matrix-based vector space language models (Turney and Pantel 2010) on the other hand, columns originally correspond to words, and counts express how often the words corresponding to the row and the column collocate in a window of some fixed length (say 5). Both in LSA and co-occurrence based VSMs, the number of contexts is at least in the thousands and gets reduced to some hundred dimensions for computation efficiency.

Neural language models (Bengio et al. 2003), on the other hand, are neural nets, trained on gigaword corpora by iterating over words in their contexts and updating some weights of the model at each word. The resulting VSMs represent similar words (types or tokens) with similar vectors, and VSMs also reflect relational similarities between words like $\mathbf{king} - \mathbf{queen} \approx \mathbf{man} - \mathbf{woman}$ (Mikolov, Yih, and Zweig 2013).

4.1 MATRIX FACTORIZATION FOR WORD MODELING

4.1.1 *Semantic differential*

Vector space models of word meaning originate with psychological research by Osgood, May, and Miron (1975). In Osgood, May, and Miron's experiments, participants were asked to scale words like *freedom* on oppositional scales like *sturdy-fragile*, be the choice simple or abstract/metaphorical. Measurements were done in several languages with great typological care, and projected from the huge place of these oppositions to a three-dimensional space by principal component analysis (PCA). The emerging inter-lingual scales called EVALUATION, POTENCY, and ACTIVITY turned out to explain much of the variation in the data. The method is called *semantic differential*. For details, see the last part of Section 2.7 in Kornai (2019).

4.1.2 *TF-IDF and PMI*

The next step in the history of VSMs has been to gain the vectors from text corpora or, in the context of information retrieval, where the method got elaborated (Salton, Wong, and Yang 1975), from text documents. Classical methods start with a *frequency matrix*, more recent ones adjust association weights in artificial neural networks, but the mathematics these systems learn turn out to be variants of each other. Turney and Pantel (2010) discuss the history of VSMs arranged by what the rows and columns of the matrices correspond to, distinguishing *term–document*, *word–context* and *pair–pattern* matrices. Each cell contains the frequency of the term (or word, ...) corresponding to the row in the document (or context, ...) corresponding to the column.

Frequencies are adjusted to balance the effect of more frequent but less informative terms, or the variation in the length of the documents. The standard *weighting* technique comes from information retrieval, where the task is to return from a pool of documents the ones that are the most relevant for (similar to) a given query. (The query is also treated as a document) This weighting is tf-idf (term frequency–inverse document frequency) scoring, but there are other methods as well.

In NLP, the information-theoretic association scores *point-wise mutual information (PMI*, Church and Hanks (1990))

$$PMI(x, y) = logP(x, y)/P(x)P(y)$$

and *positive point-wise mutual information* (PPMI, Niwa and Nitta (1994))

$$\max\{0, PMI(x, y)\}$$

became standard, and Levy and Goldberg (2014c) showed (as we will see in Section 4.2.5) that the more recent `word2vec` is mathematically equivalent to a variant of PMI, *shifted PMI*.

Besides weighting, matrices also have to be *smoothed* to reduce the amount of random noise and to fill in some of the zero elements in a sparse matrix. Semantic differential (Section 4.1.1) applies PCA, which computes word representations from the raw term–document matrix. PCA requires inverting the data matrix what became feasible for thousand-row matrices during the decades, resulting in the method of Latent Semantic Analysis, what we turn to now.

4.1.3 *Latent semantic analysis*

The main pre-neural method, which has remained an important reference point in the word embedding era (Tsvetkov, Faruqui, and Dyer 2016; Antoniak and Mimno 2018), is Latent semantic analysis (LSA, Dumais et al. (1988) and Furnas et al. (1988)) Landauer, Foltz, and Laham (1998) introduce LSA in two ways.

On the practical side, it is a method for obtaining approximate estimates of the contextual substitutability of words in text, and similarities among words and text segments. On the cognitive side, it is a model of the computational processes and representations underlying the acquisition and utilization of knowledge. While we think that it rather depends on the scientific taste of the researcher whether they motivate their work with such acquisitional claims, the practical importance of LSA in pre-embedding NLP is beyond debate. For a recent overview of LSA methods in psychology, especially author modeling, automated grading, and change over time, see Iliev, Dehghani, and Sagi (2014, Section 1.4).

Closer to the mathematical content is the way to think of LSA as representing the meaning of a word as an average of the meaning of all the passages in which it appears, and dually, the meaning of a passage as an average of the meaning of all the words it contains. The choice of dimensionality can be of great importance. LSA can be motivated in a way that the resulting dimensions may be analogous to the semantic features often postulated as the basis of word meaning, but establishing concrete relations to mentalisticly interpretable features poses daunting technical and conceptual problems. It may worth noting that LSA arrived at the same dimensionality (300), as word embeddings did (Section 4.2). The effective usage of LSA is a process of very sophisticated tuning and can be viewed as kind of art. The main factors are re-processing (stop-words, stemming), frequency matrix transformations, the choice of dimensionality, and, the choice of similarity measure. For an early sturdy of weight functions' impact, see Nakov, Popova, and Mateev (2001).

The authors point out that transformation of co-occurrence counts to log frequency divided by entropy and followed by dimensionality reduction is reminiscent of information retrieval methods, and the psycholinguistic reality of the dimensionality reduction step is often implicit and sometimes explicit in many neural net and spreading-activation architectures. The similar equivalence between word embeddings and pointwise mutual information will be discussed in Section 4.2.5.

4.1.3.1 *Singular Value Decomposition*

Data preprocessing transformations in LSA need to be described more fully. LSA subjects the data in the raw word-by-context matrix to a $\log(x + 1)$ transformation, and then each cell entry is divided by the row entropy value. The result is an estimate of the word's importance in the passage, the degree to which knowing that a word occurs provides information about which passage it appeared in.

Singular value decomposition (SVD) is the general method for linear decomposition of a matrix into independent principal components of which factor analysis is the special case for square matrices. For the reader who is not familiar with or interested in multivariate statistics, we cite Landauer, Foltz, and Laham (1998)'s elevator-pitch description of factor analysis as finding a parsimonious representation of all the intercorrelations between a set of variables in terms of a new set of abstract variables, each of which is unrelated to any other but which can be combined to regenerate the original data. SVD does the same thing for an arbitrarily shaped rectangular matrix, including the case when columns stand for words, and rows for contexts. (See the formulas in Section 4.2.8.3.) In the process, cells in the matrix originally contain the frequency. The raw cell entries f are first transformed to $\ln(1 + f)/e$ where e is the entropy of the word over all contexts. This matrix is then submitted to SVD and the — for example — 300 most important dimensions are retained (those with the highest singular values, i.e. the ones that capture the greatest variance in the original matrix). The resulting vectors of 300 real values represent each word and each context. Similarity has been usually measured by the cosine between vectors.

Related to LSA is a generative method called Latent Dirichlet Allocation (Blei, Ng, and Jordan 2003), where each document is supposed to be composed of a mixture of topics. While the dimensions of LSA may be regarded as abstract and meaningless, the dimensions in LDA correspond better to latent topics that emerge from the corpus.

4.1.4 *Relation to structuralist linguistics*

Now we summarize Sahlgren (2006), who investigates the relation between the word-space model and structuralist linguistics.

4.1.4.1 *Rethinking the distributional hypothesis: syntagma and paradigm*

The distributional hypothesis, as motivated by the works of Zellig Harris, states that differences of meaning correlate with differences of distribution, but he neither specifies what kind of distributional information we should look for, nor what kind of meaning differences it mediates.

Syntagmatic relations concern positioning, as already the Greek word *suntag-matikos* 'arranged, put in order' shows. They relate entities that co-occur in the text. They are linear, and applies to linguistic entities that occur in sequential combinations. They are combinatorial relations, which means that words that enter into such relations can be combined with each other. A syntagm is such an ordered combination of linguistic entities: written words are syntagms of letters, sentences are syntagms of words.

Paradigmatic relations, on the other hand, concern substitution. The Greek word *paradeigmatikos* means serving as a model. Saussure himself never used the word *paradigmatique*. It was Hjelmslev who coined the term as a substitute for Saussure's *associative meanings*. Paradigmatic relations are between entities that do not co-occur in the text. They hold between linguistic entities that occur in the same context but not at the same time. A paradigm is a set of such substitutable entities, usually depicted as orthogonal axes in a grid.

Although Harris was arguably more directly influenced by the works of Bloomfield than of Saussure, the latter's structuralist legacy is foundational for both Bloomfield's and Harris' theories. In Sahlgren's view, *the Saussurian refinement* of the distributional hypothesis clarifies the semantic requirements of the word-space model and the distributional methodology. A word-space model accumulated from co-occurrence information contains syntagmatic relations between words, while one from information about shared neighbors contains paradigmatic relations.

4.1.4.2 *The semantic continuum*

Sahlgren's point is that syntagmatic and paradigmatic relations between words should be discoverable by using co-occurrence information and information about shared neighbors in the word-space, respectively. *A qualitative comparison between different uses of context* e.g. in LSA (Section 4.1.3) or other models should be able to divulge the difference by empirical investigation. The author is interested in what these different uses of context entail, what their differences are, and how they can be used to build word spaces.[1]

1 While a bit irrelevant for the purposes of the present thesis, it is interesting what Sahlgren thinks about the use of a *document* as a context. Word-space algorithms that prefer a syntagmatic use of context, such as LSA, hail from the information retrieval community, where a *document* is a natural context of a word. But "document" in the sense of a topical unit is an artificial notion that hardly exists elsewhere; before the advent of library science, the idea that the content of a text could be ex-

Test	Which relation? (Is essential?)	Context
Thesaurus	both (−)	large
Association	syntagmatic (+)	small
Synonym	paradigmatic (+)	narrow
Antonym	paradigmatic (−)	wide
POS	paradigmatic (+)	narrow

Table 6: Test, relations they rely on, the degree to which the relations are essential to the test (− and +), and the context that yields the best results in the strict evaluation settings (Sahlgren 2006, Table 15.6). The thesaurus task is to list words with related meanings to the query.

Sahlgren's thesis is split to background chapters, "setting the scene" chapters, and foreground chapters, a structure we followed in that of the present thesis. The latter contain *experiments* demonstrating the differences between syntagmatic and paradigmatic uses of context: small context regions yield more syntagmatic word spaces, while narrow context windows yield more paradigmatic spaces, as can be seen in Table 6. Only a few percentage of the nearest neighbors occur in both syntagmatic and paradigmatic word spaces.

Sahlgren investigates three *parameters* of the characterization of paradigmatic contexts:

- the size of the context region,
- the position of the words within the context region, and
- the direction in which the context region is extended. The only experiment he was aware of exploiting the directional information in a words-by-words co-occurrence matrix was Schütze (1993).

In his experiments, Sahlgren compares different *weighting schemes* of the slots for the paradigmatic uses. The two extremes are constant weighting over the window, and aggressive distance weighting according to the formula 2^{1-l}, where l is the distance to the focus word. Possibilities in between include linear distance weighting and $1/l$.

In the concluding chapter, Sahlgren answers his research questions. Is it at all possible to extract semantic knowledge by merely looking at usage data? Clearly, yes. Does the word-space model constitute a complete model of the full spectrum of meaning, or does it only convey specific aspects of meaning? It is complete as far as is reflects a

pressed with a few index terms must have seemed strange. In the "real" world, content is something we reason about, associate to, and compare. In the world beyond information-retrieval, text is a continuous flow where topics intertwine and overlap and the notion of a "document" is at best an arbitrary choice. In a whole document nearly every term can co-occur with every other.

structuralist dichotomy of syntagma and paradigm. If we believe that meaning is essentially referential, then no.

4.1.4.3 *"Future" work*

The future work section lists problems related to which much has been achieved since 2006, but they still remain major problems. One is that word spaces may have (i) a common internal structure that can be utilized to differentiate between different types of relations within the word space; and (ii) a discoverable "latent" dimensionality. While compositionality is not without controversy in the philosophy of language, word-space models may be extended to handle phrase, clause, sentence, paragraph, "document" and text level meaning too. The word-space model may have the flexibility and ability to continuously evolve when subjected to a continuous data flow.

Finally, Sahlgren remarks that the word-space model is not a psychologically realistic model of human semantic processing. It is arguable that humans also use extra-linguistic context when learning, understanding, and using language. The inability to reach beyond the limits of textuality is the most disqualifying feature of the word-space model with regards to the referential aspect of meaning.

4.1.5 *A compression-based method*

Cilibrasi and Vitányi (2004) present a similarity measure between words and phrases based on information distance and Kolmogorov complexity, using Google page counts. In the Turney and Pantel (2010) classification, this is a term–document model. This similarity measure is the special case of a compression-based universal similarity metric among objects given as finite binary strings. These strings include genomes, music pieces in MIDI format, computer programs, pictures in simple bitmap formats, or time sequences such as heart rhythm. The universal metric is feature-free in the sense that it does not look for particular features, but analyzes all features simultaneously and determines the similarity between every pair of objects according to *the most dominant shared feature*. The word similarity measure is based on "the Google semantics of a word or phrase", i.e. the set of web pages returned by the query concerned.

They normalize the introduced distance to make it relatively stable with respect to the index size (Normalized Google Distance, NGD). The NGD of *horse* and *rider* is 0.443. The distance is usually between 0 (identical) and 1 (unrelated), but not always (see below). If the distance is calculated from the index of only one-half of the pages, this distance only deviates to 0.460.

A drawback of the Google semantics is that terms with different meaning may have the same semantics, especially *opposites* often have a similar semantics. The paper offers more literature (of course, from

before 2005) on how representative Google hits are for language. The theoretical underpinning is based on the theory of Kolmogorov complexity, in terms of coding and compression. The NGD formula

$$NGD(x,y) = \frac{G(x,y) - \min(G(x), G(y))}{\max(G(x), G(y))}$$
$$= \frac{\max(\log f(x), \log f(y)) - \log f(x,y)}{\log N - \min(\log f(x), \log f(y))}$$

is similar to many earlier formulas in this area, but not equivalent to any of them.

It has to be noted that the returned *Google counts are approximate*. The situation used to get worse if one used the boolean OR operator between search terms, but the measure is based on the AND operator, which is less problematic. When the paper was written, Google already estimated the number of hits based on samples, and the number of indexed pages already changed rapidly. To compensate for the latter effect, the authors have inserted a normalizing mechanism. Web searches for rare two-word phrases correlated well with frequency in traditional corpora, as well as with human judgments.

4.1.5.1 *Kolmogorov complexity, information distance, compression-based similarity*

Information can be compressed to different extents. The Kolmogorov complexity $K(x)$ is the length, in bits, of the ultimate compressed version from which x can be recovered by a general decompression program. An earlier paper considered the following information distance $E(x,y)$: given two strings x and y, what is the length of the shortest binary program in the reference universal computing system such that the program computes output y from input x, and also output x from input y. Up to a negligible logarithmic additive term, $E(x,y) = K(x,y) - \min K(x), K(y)$, where $K(x,y)$ is the binary length of the shortest program that produces the pair x, y and a way to tell them apart. This distance $E(x,y)$ is actually a metric.

E is *universal* for the family of computable distances, i.e. E minorizes every admissable distance up to an additive constant, where admissible means nonnegative, symmetric, and computable. More intuitively, this means that the information distance determines the distance between two strings minorizing the dominant feature in which they are similar. This measure has to be normalized, because if small strings differ by an information distance which is large compared to their sizes, then the strings are very different. The normalized information distance (NID) has values between 0 and 1, and it is universal: minorizes, up to a vanishing additive term, every other possible normalized computable distance. The NID is uncomputable since the Kolmogorov complexity is uncomputable, but we can use real data compression programs to approximate the Kolmogorov complexities $K(x)$, $K(y)$, $K(x,y)$.

4.1.5.2 *Google distribution, Normalized Google Distance, and their universality*

In the third section the authors show that the Google distribution is universal for all the individual web users distributions. We cannot use the probability of the events directly to determine a prefix code, or, rather the underlying information content implied by the probability because events overlap and hence the summed probability exceeds 1. But absolute probabilities allow us to define the associated prefix code-word lengths (information contents) for both the singletons and the doubletons. Let G denote the prefix-code word length defined from the relative frequency of the hits.

The Google Similarity Distance has the following properties:

- The range of the NGD is basically in between 0 and ∞. More precisely, it is slightly negative if the Google counts are untrustworthy and state $f(x, y) > \max\{f(x), f(y)\}$.
- If $f(x) = f(y) = f(x, y) > 0$, then $NGD(x, y) = 0$.
- If frequency $f(x) = 0$, then for every search term y we have $NGD(x, y) = \infty/\infty$, which we take to be 1 by definition.
- NGD is always nonnegative and $NGD(x, x) = 0$ for every x.
- NGD is symmetric ($NGD(x, y) = NGD(y, x)$).
- The NGD does not satisfy the triangle inequality, i.e. NGD is not a metric.

The paper includes clustering and classification (against WordNet) experiments to validate the universality, robustness, and accuracy of the proposal.

4.1.6 *Mathematical processing*

Now we summarize Turney and Pantel (2010, Section 4)'s discussion of the mathematical processing for distributed word models. This will be especially important in Chapter 7.

First the frequency matrix is built by scanning sequentially through the corpus, and recording events and their frequencies in a hash table, a database, or a search engine index. The frequency matrix has to be represented in a sparse way (i.e. most items are 0).

4.1.6.1 *Weighting the Elements*

The weights of the elements in the matrix have to be adjusted, because common words will have high frequencies, yet they are less informative than rare words. Infomation retrieval uses the tf-idf (term frequency $\times$

inverse document frequency) family of weighting functions, where an element gets a high weight when the corresponding term is frequent in the corresponding document (i.e. tf is high), but rare in other documents in the corpus (i.e. df is low). Document length has to be normalized.

Affixation, especially derivational affixation is problematic both from linguistic and computation point of view. The linguistic problem is to delineate the inventory of compositional affixes. The compositional problem is that though different forms of the same lexeme are correlated, yet we may not want to lemmatize them, because they may have slightly different meanings. An idea that did not become standard is to reduce the weights of derivatives when they co-occur in a document.

A key step in pre-neural machine learning was feature selection. One of the most popular word association scores remains Pointwise Mutual Information, which we will discuss in detail in Section 7.2.

4.1.6.2 *Smoothing the Matrix*

The goal of smoothing the matrix is to reduce the amount of random noise and to fill in some of the zero elements that are due to data sparsity. The other direction, sparsification is a hot topic today (Sanh et al. 2019), but it goes beyond the limits of this thesis. The mathematical method of truncated (or thin) Singular Value Decomposition (SVD) is standardly applied to either document similarity (Latent Semantic Indexing), or word similarity (Latent Semantic Analysis, Section 4.1.3).

SVD decomposes X into the product of three matrices $U\Sigma V^\intercal$, where U and V are in column orthonormal form (i.e. the columns are orthogonal and have unit length, $U^\intercal U = V^\intercal V = I$), and Σ is a diagonal matrix of singular values. If X is of rank r, then Σ is also of rank r. Let Σ_k, where $k < r$, be the diagonal matrix formed from the top k singular values, and let U_k and V_k be the matrices produced by selecting the corresponding columns from U and V. The matrix $U_k\Sigma_k V_k^\intercal$ is the matrix of rank k that best approximates the original matrix X, in the sense that it minimizes the approximation errors. That is, $\widehat{X} = U_k\Sigma_k V_k^\intercal$, which is called the *truncated SVD*, minimizes $|\widehat{X} - X|_F$ over all matrices $\widehat{X}$ of rank k, where $|\dots|_F$ denotes the Frobenius norm.

The authors list four aspects of what SVD is looking for: latent meaning, noise reduction, indirect or high-order co-occurrence (when two words appear in similar contexts), or sparsity reduction. Truncated SVD implicitly assumes that the vectors have a Gaussian distribution – Minimizing the Frobenius norm $|\widehat{X} - X|_F$ will minimize the noise, if the noise has a Gaussian distribution – but this assumption is not satisfied by word frequencies.

4.1.6.3 *Comparing the Vectors*

There are many different ways to measure the similarity of two vectors, but the most popular one is clearly cosine similarity, while the most

intuitive one remains the Euclidean distance. In classical information retrieval, it has been commonly said that, properly normalized, the difference in retrieval performance using different measures is insignificant. Distances include the Manhattan distance, or, from information theory, Hellinger, Bhattacharya, and Kullback-Leibler. Dice $2xy/(|x|^2 + |y|^2)$ and Jaccard have set-theoretic motivation.

Lee (1999) gives the principle that measures that focused more on overlapping coordinates and less on the importance of negative features (i.e. coordinates where one word has a nonzero value and the other has a zero value) appear to perform better. In her experiments, the Jaccard, Jensen-Shannon, and L1 measures seemed to perform best.

Other researchers studied the linguistic and statistical properties of the similar words returned by various similarity measures and found that the measures can be grouped into three classes: high-frequency sensitive measures, low-frequency sensitive measures, similar-frequency sensitive methods. Given a word w_0, if we use a high-frequency sensitive measure to score other words w_i according to their similarity with w_0, higher frequency words will tend to get higher scores than lower frequency words. If we use a low-frequency sensitive measure, there will be a bias towards lower frequency words. Similar-frequency sensitive methods prefer a word w_i that has approximately the same frequency as w_0.

4.1.6.4 *Efficient comparisons*

One section in Turney and Pantel (2010) discusses methods like distributed sparse matrix multiplication and Random Indexing. Randomized algorithms are based on the idea that high-dimensional vectors can be randomly projected into a low-dimensional subspace with relatively little impact on the final similarity scores. Random Indexing (RI) is an approximation technique that computes the pairwise similarity between all rows (or vectors) of a matrix. There are index vector elements of which are mostly zeros with a small number of randomly assigned $+1$'s and -1's. The cosine measure between two rows r_1 and r_2 is then approximated by computing the cosine between two fingerprint vectors, fingerprint(r_1) and fingerprint(r_2), where fingerprint(r) is computed by summing the index vectors of each non-unique coordinate of r.

Locality sensitive hashing (LSH, Broder, 1997) is similar technique. LSH functions include the the Min-wise independent function, which preserves the Jaccard similarity between vectors, and functions that preserve the cosine similarity between vectors.

4.2 NEURAL WORD EMBEDDINGS

4.2.1 *Symbolic structures in connectionism*

As a thesis submitted to a theoretical linguistics programme, this work may start its account of neural word models with Rumelhart and McClelland (1986), one of two papers from the same year in which distributed representation of words was proposed. (The other one is Hinton, McClelland, and Rumelhart (1986)). Rumelhart and McClelland's paper belongs to the infamous past tense debate. However, we prefer taking our ideological heritage from Smolensky (1990, Section I), what we summarize now.

4.2.1.1 *Discrete and continuous computations*

Connectionist models rely on parallel numerical computation rather than the serial symbolic computation of traditional artificial intelligence (AI) models. Smolensky argues that connectionist models will offer an opportunity to escape the brittleness of symbolic AI systems, and develop more human-like intelligent systems, but only if we can find ways of naturally instantiating the sources of power of symbolic computation within fully connectionist systems. The connectionist approach, on the one hand, is an excellent opportunity for formally capturing the subtlety, robustness, and flexibility of human cognition, and for elucidating the neural underpinnings of intelligence. The symbolic approach, on the other, has provided tremendous insights into the nature of the problems that must be solved in intelligent systems, and of techniques for solving these problems.

The paper is part of an effort to extend the connectionist framework to naturally incorporate symbolic computation, without losing the virtues of connectionist computation; i.e. integrate the discrete mathematics of symbolic computation and the continuous mathematics of connectionist computation. Language can be represented by objects like a phrase-structure tree, or even as a simple sequence of words. The representation problem is characterized as finding a mapping from the set of structured objects to a vector space.

Smolensky takes an analogy from mathematics: representing abstract groups as collections of linear operators on a vector space. Discrete group theory and the continuous vector space theory interact, and this relation extends to applications like quantum physics. In physics, elementary particles involve a discrete set of particle species which exhibit many symmetries, that are described by group theory. Yet underlying elementary particle state spaces are continuous.

In human language processing, the discrete symbolic structures that describe linguistic objects are actually "imbedded" in a continuous connectionist system that operates on them with flexible, robust processes that can only be *approximated* by discrete ones. Smolensky refer to

structures as symbolic ones, because the principal cases of his interest are objects like strings and trees, however, his analysis is of structured objects in general; it applies equally well to objects like images and speech trains. (His view is not that mental operations are always serial symbol manipulations, but that the information processed often has useful symbolic descriptions.)

Smolensky seeks a fully distributed representation in which each output neuron participates in the representation of many different outputs. In the tensor product representation he proposes, both the variables and the values can be arbitrarily nonlocal, enabling (but not requiring) representations in which every unit is part of the representation of every linguistic constituent in the structure. The representation can be used recursively, and connectionist representations of operations on symbolic structures and recursive data types, can be naturally analyzed.

4.2.1.2 *Why inject symbolic structure in a neural network?*

The motivation for pursuing the representation of symbolic structures in connectionist systems lies in the connectionist modeling of higher cognitive processes such as language. Here the central question is: What are computationally adequate connectionist representations of strings, trees, and sentences? The essence of the connectionist approach, people might say, is to expunge symbolic structures from models of the mind. But a reasonable starting point is to take linguistic analysis of the structure of linguistic objects seriously, and to find a way of representing this structure in a connectionist system: it is important to find adequate connectionist representations of these trees or strings. The authors' hope is that new connectionist representations of linguistic structures will rest on prior understanding of connectionist representations of existing symbolic descriptions of linguistic structure. The importance of representing linguistic structures exceeds NLP: these representations are the basis for connectionist models of conscious, serial, rule-guided behavior: all higher thought processes.

One argument against designing a connectionist representation of symbolic structures goes like this: Just as a child somehow learns to internally represent sentences with no explicit instruction on how to do so, so a connectionist system with the right learning rule will somehow learn the appropriate internal representations; The problem of linguistic representation is not to be solved by a connectionist theorist but rather a connectionist network. Smolensky's response is the following:

- In the short term, at least, our learning rules and network simulators do not seem powerful enough for unstuctured learning,
- we will still need to explain how the representation is done,
- we should build bridges as soon as possible between accounts of language; the problem is just too difficult to start all over again from scratch,

- to experiment now with connectionist learning of rather complex skills (e.g. parsing, anaphoric resolution, and semantic interpretation, all in complex sentences), we need connectionist representation of the input and output. We want to study the learning of the operations without waiting for the discovery of the linguistic representations.

- Language is more than just a domain for building models: it is a foundation on which the entire traditional theory of computation rests. It is crucial for how the basic concepts of symbolic computation and formal language theory relate to connectionist computation.

4.2.2 *Neural language modeling*

At least before the neural revolution in NLP, the term *language modeling* was restricted to the task of "predicting the next word", which is equivalent to computing the probability (naturalness) of a word sequence. Probabilities are estimated using (relative) frequencies. As there are infinitely many possible sentences but the model is trained on a finite sample, the main point is in generalization. A simple and effective approach to language modeling is the family of n-gram models (Brown et al. 1992) that make the Markov assumption, i.e. the simplifying assumption that the probability of a word in a context depends only on preceding words of some fixed number (four in most applications of the time). Thus the probability of the Hungarian word string *minden madár társat választ* ('every bird is choosing a mate')[2] is computed as

$$P(\texttt{\^{} minden madár társat választ \$}) =$$

$$P(\texttt{ minden } | \texttt{ \^{} }) \cdot P(\texttt{ madár } | \texttt{ minden }) \cdot P(\texttt{ társat } | \texttt{ madár }) \cdot$$
$$\cdot P(\texttt{ választ } | \texttt{ társat }) \cdot P(\texttt{ \$ } | \texttt{ választ })$$

$P(\texttt{ madár } | \texttt{ minden })$ denotes the probability of the word *madár* given that the preceding word was *minden*. `^` and `$` denote the beginning and the end of the string, respectively. While n-gram models are easy to understand and useful in application, they have the disadvantage of not capturing morphological and semantic relations between words. This is the problem that the neural language model (Bengio et al. 2003) solved.

Bengio et al. (2003) implement the n-gram language model relying on shared-parameter multi-layer neural networks. Their network has millions of parameters, and it is trained on tens of millions of examples. Training such large-scale model is expensive but feasible, scales to large contexts, and yields good comparative results.

The idea of fighting the so called *curse of dimensionality* with distributed representations is summarized by the authors as associating

2 This sentence is from the song that gave the title of the conference where Makrai (2014) was published.

with each word in the vocabulary a distributed word feature vector (a real-valued vector in R^m); expressing the joint probability function of word sequences in terms of the feature vectors of these words in the sequence; and learning simultaneously the word feature vectors and the parameters of the probability function. The objective can be the log-likelihood of the training data or a regularized criterion, e.g. by adding a weight decay penalty i.e. like in ridge regression, the squared norm of the parameters as a penalty.

The paper cites rich related work for the idea of using neural networks to model high-dimensional discrete distributions and, from the early days of connectionism, the idea of learning a distributed representation for symbolic data. In their view, neural networks for language modeling are not new either with work in character-level LM based neural text compression with or without hidden units and a single or more input words. What is more well known, generalization from training sequences have been obtained in the form of similarities between words: clusterings of the words with words associated deterministically or probabilistically with classes. Vector-space representation for words has been well exploited in the context of an n-gram based statistical language model, using LSI to dynamically identify the topic of discourse. Finally, vector-space representation for symbols in the context of neural networks, and especially a parameter sharing layer, has been pioneered in text-to-speech mapping.

Bengio et al. (2003) is the kind of paper whose future work section forecast the most important steps of the next 10-15 years, especially hierarchical softmax (Morin and Bengio 2005), the recurrent language model Mikolov (2010), negative sampling (Mikolov, Sutskever, et al. 2013), "interpreting (and possibly using) the word feature representation" (Mikolov, Yih, and Zweig 2013), and sub-word encoding (Bojanowski et al. 2017). A section sketches an energy-based extension.

4.2.3 *Unsupervised pre-training and noise-contrastive estimation*

One of the key components of the NLP advances in the last decade is parameter sharing in the form of unsupervised pre-training introduced by Collobert et al. (2011), who train a single model for tasks including part-of-speech tagging, chunking, named entity recognition, and semantic role labeling. The system learns internal representations based on vast amounts of mostly unlabeled training data. This representation is then used as a basis for building a freely available tagging system with good performance. The architecture is similar to Bengio et al. (2003)'s language model discussed in the previous section, but it uses noise contrastive estimation to spare the computation of the normalization term needed for probabilistic modeling. A couple of years later, noise-contrastive estimation, or simply negative sampling, became an ingredient of the very influential skip-grams model we will see in the

next section. Besides its great importance in the development of VSMs, Collobert et al.'s work has also relevant in this thesis because we used their vectors in Sections 5.2, 5.4 and 5.5.

This work is also one of the most remarkable linguistic applications of one of the major neural architectures, *convolution*, which was originally invented for computer vision. The window approach described so far performs well for most NLP tasks Collobert et al. choose, but it fails with semantic role labeling (SRL), where the predicate may fall outside the window. This task requires the consideration of the whole sentence. Among the main neural networks architectures, one of the natural choices to tackle this problem in a convolutional networks.

A convolutional network is a sequence of alternating convolutional and pooling layers. A convolutional layer is a generalization of a window approach: given a sequence represented by columns in a matrix, a matrix-vector operation is applied to each window of successive windows in the sequence, where the weight matrix is constant across all windows. Convolutional layers extract local features around each window, and they are often stacked to extract higher level features.

The size of the output of the convolutional layer depends on the number of words. Local feature vectors extracted by the convolutional layers have to be combined to obtain a global feature vector, with a fixed size, in order to apply subsequent layers. Traditional convolutional networks often apply a (possibly weighted) average or a max operation over "time". Average does not make much sense in the SRL case, as in general most words in the sentence do not have any influence on the semantic role of a given other word. So the authors used a max approach. The network finally produces one score per possible tag for the given task, as in the window approach.

4.2.4 *word2vec*

Deeper in its effect on the broad NLP community than in its architecture, the first wave of the neural revolution has been pre-trained *word embeddings*, word models learned by shallow neural networks in an unsupervised way, which have become very popular since Mikolov, Sutskever, et al. (2013), who implemented a log-bilinear model to learn continuous representations of words on very large corpora efficiently. These more accurate variants of earlier VSMs, map "similar" word to similar vectors in a space of some hundred dimensions. Word similarity covers syntax and semantics, and vector similarity is mostly measured by cosine similarity. Embeddings also reflect analogical quadruples (Mikolov, Yih, and Zweig (2013), Section 5.1) like

$$\mathbf{woman} - \mathbf{man} \approx \mathbf{queen} - \mathbf{king}$$

Mikolov, Le, and Sutskever (2013) discovered that VSMs of different languages have such similarities that a linear mapping can map the representations of words in a source language to the representation of their translations, see Section 8.5 for details.

Most of the main contributions of this thesis are related to the `word2vec` line of research. Section 5.2 offers a Hungarian equivalent of the analogical test set, Section 5.3 compares word embeddings based on the dictionary induction method by Mikolov, Le, and Sutskever (2013), Sections 5.4 and 5.5 investigate two lexical relations with the vector offset method of Mikolov, Yih, and Zweig (2013), and the linear translation method is the basis for our inquiry to polysemy in Chapter 8.

4.2.5 *Word embeddings as matrix factorization*

The series of papers Levy and Goldberg (2014c), Goldberg and Levy (2014), Levy and Goldberg (2014b), Levy, Goldberg, and Dagan (2015), and Levy et al. (2015) unfolded the series Mikolov, Chen, et al. (2013), Mikolov, Sutskever, et al. (2013), Mikolov, Yih, and Zweig (2013), Mikolov, Le, and Sutskever (2013), and Le and Mikolov (2014) as Zhuangzi unfolded Laozi. As we have already cited, Levy and Goldberg (2014c) showed that skip-gram with negative-sampling (SGNS) is implicitly factorizing a word-context matrix,

$$w \cdot c = \mathrm{PMI}(w, c) - \log k$$

whose cells are the pointwise mutual information (PMI) of the respective word and context pairs, shifted by a global constant. Similarly, an embedding model based on noise-contrastive estimation (Mnih and G. E. Hinton 2008) was shown to be implicitly factorizing a similar matrix, where each cell is the (shifted) log conditional probability of a word given its context. SGNS is much less sensitive to extreme and infinite values than the pure SVD of a PPMI matrix, due to a sigmoid function surrounding $w \cdot c$, and the weighting function: rare (w, c) pairs affect the objective much less.

Levy and Goldberg (2014c) improved results on standard test sets of the time, two word similarity tasks and one of two analogy tasks, with a sparse Shifted PPMI word-context matrix representation of the words. (We introduced PPMI in Section 4.1.2.) They also showed that dense low-dimensional vectors from exact factorization with SVD provides at least as good as SGNS's solutions for word similarity tasks. On analogy questions, SGNS remains superior to SVD. They conjectured that this stems from the weighted nature of SGNS's factorization.

4.2.6 *Global optimization*

The interest in why SGNS can capture such fine-grained semantic and syntactic regularities using vector arithmetic inspired an other implementation, *GloVe* (Pennington, Socher, and Manning 2014), which, besides its mathematical elegance, apparently became most frequently applied word embedding, probably more frequently the the original set by Mikolov et al. Our experiments in Chapters 5 and 8 are no exception. The abbreviation states for global vectors or, more precisely, globally optimized vectors. The authors claim that models, such as SGNS, that train on separate local context windows instead of on global co-occurrence counts, poorly utilize the statistics of the corpus. The global approach is made possible by training only on the nonzero elements in the word-word co-occurrence matrix.

The basis of GloVe is the logbilinear model

$$w_i^\intercal \widehat{w}_k + b_i + \widehat{b}_k = \log(X_{ik}),$$

where X is the co-occurrence matrix, w and $\widehat{w}$ are the focus and context vectors for each word, and b and $\widehat{b}$ are bias vectors. The two kinds of vectors w and $\widehat{w}$ are needed because words rarely appear in their own context, but we do not want $w^\intercal w$, the squared norm of w, to be small.

The objective above is approximated with weighted least-squares regression, where the weighting is motivated by that rare co-occurrences are noisy and carry less information than the more frequent ones. They introduce the weighting function $f(X_{ij})$, where

$$f(x) = \begin{cases} (x/x_{\max})^\alpha & \text{if } x < x_{\max} \\ 1 & \text{otherwise,} \end{cases}$$

with $x_{\max} = 100$ and $\alpha = 3/4$. Word pairs with a co-occurrence below $x_{\max}$ are downweighted (by a slightly concave function). It is interesting that a similar fractional power scaling was found to give the best performance in Mikolov, Chen, et al. (2013).

Levy, Goldberg, and Dagan (2015) point out that if we were to fix

$$b_w = \mathit{log}\text{freq}(w) \text{ and}$$

$$b_c = \mathit{log}\text{freq}(c),$$

this would be almost equivalent to factorizing the PMI matrix shifted by $\mathit{log}(|D|)$, where $|D|$ is the vocabulary size. However, GloVe learns these parameters, giving an extra degree of freedom over SVD and SGNS. (Unlike Arora et al. (2015)'s RandWalk model, which has a linear relation between the squared norms of the word vectors and the logarithm of the word frequencies.)

Pennington, Socher, and Manning (2014) compare their method to `word2vec` mathematically and in performance in their sections 3.1 and 4.7, respectively. The quantitative comparison is complicated by many parameters that have a strong effect on performance. They control for the main sources of variation, vector length, context window size, corpus, and vocabulary size. The most important remaining variable to control for is *training time*.

For GloVe, the relevant parameter is the number of training iterations, while for `word2vec`, the obvious choice would be the number of training epochs, but back then the code was restriced to a single epoch. They measure training time instead by the number of negative samples, which effectively increases the number of training words seen by the model. For the same corpus, vocabulary, window size, and training time, GloVe consistently outperforms `word2vec`. More interestingly from the big-picture perspective, `word2vec`'s performance decreases if the number of negative samples increases beyond about 10.

4.2.7 *Word analogies, direction, and multiplication*

Levy and Goldberg (2014b) generalize word analogies as searching for a word that maximizes a linear combination of three pairwise word similarities

$$\arg\max_{b^*}(\mathrm{sim}(b^*, b - a + a^*)) = \arg\max_{b^*}(\cos(b^*, b - a + a^*))$$

$$= \arg\max_{b^*}(\cos(b^*, b) - \cos(b^*, a) + \cos(b^*, a^*))$$

(e.g. $b = \text{king}, a = \text{man}, a^* = \text{woman}, b^* = \text{queen}$), and show that the linear representation of lexical properties is not restricted to neural word embeddings: a similar amount of relational similarities can be recovered from traditional distributional word representations. Calling the original additive objective 3CosAdd, they introduce PairDirection, which requires only the direction of $a^* - a$ to be conserved by $b^* - b$, and the multiplicative variant 3CosMul

$$\arg\max_{b^*} \frac{\cos(b^*, b) \cdot \cos(b^*, a^*)}{\cos(b^*, a) + \varepsilon}.$$

$\varepsilon = 0.001$ is used to prevent division by zero. Though it was not mentioned in the paper, Mikolov, Yih, and Zweig (2013) used PairDirection for solving the semantic analogies of the SemEval task, 3CosAdd for solving the syntactic analogies.

PairDirection performs very well on multiple choice tasks, yet very poorly on full vocabulary searches. The difference is attributed to that PairDirection is likely to find candidates b^* that have the same relation to b as reflected by $a - a^*$ but these candidates are not necessarily

similar to b. In the *queen* example, PAIRDIRECTION may return feminine entities, but not necessarily royal ones. The motivation for 3COSMUL is to avoid the "soft-or" behavior of linear objectives, i.e. that they allow one sufficiently large term to dominate the expression.

4.2.8 *Improving PPMI-SVD with neural lessons*

Levy, Goldberg, and Dagan (2015) improve traditional distributional similarity models with lessons learned from word embeddings. We will build in this line of research especially in Chapter 7. Their experiments reveal that much of the performance gains of word embeddings are due to certain system design choices and hyper-parameter optimizations. By making the hyper-parameters explicit, the authors show how they can be adapted and transferred into the traditional count-based approach. Changing the setting of a single hyper-parameter yields more than switching to a better algorithm or training on a larger corpus.

For historical reasons (Baroni, Dinu, and Kruszewski 2014), they refer to PPMI and SVD as "count-based" and to SGNS and GloVe as "neural" or "prediction-based". The following hyper-parameters can be transferred from `word2vec` and GloVe to count-based methods:

4.2.8.1 *Pre-processing Hyperparameters*

Words can be weighted according to their distance from the focus word. In traditional count-based methods, it is less common, but also explored (Sahlgren (2006), Section 4.1.4.2). GloVe uses $1, 1/2, 1/3, \ldots$, and `word2vec` $w/w, w-1/w, \ldots$. What seem important is the *dynamic context window*: `word2vec` implements its weighting scheme by uniformly sampling the actual window size between 1 and L.

Subsampling is for diluting very frequent words. Mikolov, Chen, et al. (2013) randomly remove words that are more frequent than some threshold t. While `word2vec`'s code implements a slightly different formula, Levy, Goldberg, and Dagan followed the formula presented in the original paper (equation 2). Subsampling in `word2vec` is dirty in the sense that the removal of tokens is done before the corpus is processed into word-context pairs. Levy, Goldberg, and Dagan found the impact of dirty and clean subsampling comparable, and report dirty.

Finally, `word2vec` removes some rare words before creating context windows, but Levy, Goldberg, and Dagan's experiments showed that the effect of this was small.

4.2.8.2 *Association Metric Hyperparameters*

The authors define Shifted PMI as

$$SPPMI(w,c) = max(PMI(w,c) - log(\mathrm{k}), \mathrm{0})$$

k has two distinct functions:

- to better estimate the distribution of negative examples: a higher k means more data and better estimation, and
- it affects the probability of observing a positive example: a higher k means that negative examples are more probable.

Shifted PPMI captures only the second aspect of k. They experiment with three values of k: 1, 5, 15.

Finally, in `word2vec`, negative examples (contexts) are sampled according to a *smoothed* unigram distribution. Smoothing alleviates PMI's bias towards rare words.

4.2.8.3 *Post-processing Hyperparameters*

When word vectors are used in some downstream task (an intrinsic test or a real application), *context vectors* c are often *added* to focus vectors w. This was originally motivated as an ensemble method. While this addition does not apply to PPMI, it is interesting that they authors provide a different interpretation of its effect: it adds first-order similarity terms to the second-order similarity function. Second-order similarity $w_x \cdot w_y, c_x \cdot c_y$ measures the extent to which the two words are replaceable based on their tendencies to appear in similar contexts, and are the manifestation of Z. S. Harris (1954)'s distributional hypothesis. First-order similarity $w_x \cdot c_y$, on the other hand, is the tendency of one word to appear in the context of the other.

Recall that truncated Singular Value Decomposition (SVD) is a common method of dimensionality reduction, which finds the optimal rank d factorization with respect to L_2 loss. SVD has been popularized in NLP via Latent Semantic Analysis (LSA, Deerwester, Dumais, and Harshman (1990), Section 4.1.3). The word-context matrix M is factorized as

$$M = U \cdot \Sigma \cdot V$$

where U and V are orthonormal and Σ is a diagonal matrix of eigenvalues. The representations are obtained as $W_{SVD} = U_d \cdot \Sigma_d$ for words and $C_{SVD} = V_d$ for contexts.

In the SVD-based factorization, the context matrix C_{SVD} is orthonormal while the word matrix W_{SVD} is not. The factorization by SGNS's is much more "symmetric": neither W_{w2v} nor C_{w2v} is orthonormal, and there is no bias to either of the matrices in the training objective. Symmetry can be achieved in SVD by *weighting the eigenvalue matrix* Σ_d with the exponent p, what has a significant effect on performance, and should be tuned. The final hyper-parameter of any vector space language model is whether rows and/or columns are normalized.

4.2.8.4 *Low-dimensional embeddings and isotropy*

Arora et al. (2016) emphasizes that $\langle v_w, v_{w'} \rangle \approx PMI(w, w')$ was only true if there were no dimension constraints, but, in practice, low-dimensional

embedding are used. They argue that the low dimensionality of word embeddings plays a key role. In previous papers, the model is agnostic about the dimension of the embeddings, and the superiority of low-dimensional embeddings is an empirical finding (starting with Deerwester, Dumais, and Harshman (1990)). Arora et al.'s theoretical analysis makes the key assumption that the set of all word vectors (which are latent variables of the generative model) are spatially isotropic, i.e. they have no preferred direction in space. Having n vectors be isotropic in d dimensions requires $d \ll n$. This is related to the emergence of the "relations = lines" phenomenon.

4.2.9 *What's in a similarity score?*

The basic evaluation for static word embeddings has been in word similarity, but the method has many shortcomings. Now we summarize Avraham and Goldberg (2016) to illustrate these. Avraham and Goldberg redesign the annotation task to achieve higher inter-rater agreement, and propose a performance measure which takes the reliability of each annotation decision in the dataset into account.

Datasets for Word Similarity Evaluation have been standardly used with rank correlation (Spearman's ρ). Hill, Reichart, and Korhonen (2015) pointed out that in some datasets, associated but dissimilar words, e.g. $\langle$*singer*, *microphone*$\rangle$, ranked high, sometimes even above pairs of similar words. Hill, Reichart, and Korhonen also found a clear preference for hyponym-hypernym pairs, e.g. $\langle$*cat*, *pet*$\rangle$ and $\langle$*winter*, *season*$\rangle$ over cohyponyms pairs like $\langle$*cat*, *dog*$\rangle$ (and, less outrageously, over antonyms pairs $\langle$*winter*, *summer*$\rangle$).

Avraham and Goldberg summarize the problems as follows:

- The rating scales are vulnerable to a variety of biases. This problem was earlier addressed by asking the annotators to rank each pair in comparison to 50 randomly selected pairs, but that resulted in a daunting annotation task.

- Different relations are rated on the same scale. A difference of 1.8 similarity scores can testify to anything from no difference, e.g. $\text{sim}(\text{smart}, \text{dumb}) = 0.55$, $\text{sim}(\text{winter}, \text{summer}) = 2.38$, to true superiority of one pair, e.g. $\text{sim}(\text{cab}, \text{taxi}) = 9.2$, $\text{sim}(\text{cab}, \text{car}) = 7.42$..

- Different target words are rated on the same scale. Even within pairs in a targeted relation, there are ill-defined comparisons, e.g.: $\langle$*cat*, *pet*$\rangle$ vs. $\langle$*winter*, *season*$\rangle$. Pairs which share the target are much more natural to compare, e.g. the comparison $\langle$*cat*, *pet*$\rangle$ vs. $\langle$*cat*, *animal*$\rangle$ is natural. Penalizing a model for preferring $\langle$*cat*, *pet*$\rangle$ over $\langle$*winter*, *season*$\rangle$ or vice versa impairs the evaluation reliability.

- The evaluation measure does not consider annotation decisions reliability. Reliability should be determined by the agreement of the annotators.

They published two datasets of Hebrew nouns with the following features:

- The annotation task is an explicit ranking task: each pair is directly compared with a subset of the other pairs, but, unlike in earlier work, with only a few carefully selected pairs, following the principles above.
- Only pairs in a single preferred relation type (hyponym-hypernym in one dataset, and cohyponym in the other one) are presented to the annotators, what spares the annotators the effort of considering the type of the similarity, and lets them concentrate on the strength of the similarity.
- Any pair is be compared only with pairs sharing the same target word.
- The dataset includes a reliability indicator with a probabilistic interpretation.

4.2.10 *Retrofitting vectors to semantic lexicons*

The two main topics of this thesis are semantic networks (relational representations of lexical meaning) and neural word embeddings. The original goal of both have been to model assotiations in the human mind that make linguistic processing possible. Early research in computational linguistics was based on manual implementation of expert knowledge, and hand-crafted tools remain useful even today. Since the nineties, computers have become able to learn from text corpora of increasing size, and in recent years, artificial neural networks became state-of-the-art in many computational applications, but their interpretability remains poor. In this section, we investigate methods of injecting knowledge from semantic networks to (static) word embeddings.

Work before Faruqui et al. (2015) either augmented the co-occurrence matrix in a relation-specific way, or changed the objective of the word vector training algorithm to include some relational knowledge. The latter involves enhancing `word2vec` to include more similarity knowledge or word relational knowledge and or latent semantic analysis for antonym specific polarity induction or multi-relational knowledge. These methods are limited to particular vector models. Faruqui et al. introduced a graph-based learning technique. The training objective includes an additional term for new vectors to be similar to the vectors of related word types. Relations are taken from semantic lexicons such as

WordNet (Section 2.4.1), FrameNet (Section 2.4.2), and the Paraphrase Database.

Besides the English GloVe, skip-gram with hierarchical softmax, and the multi-prototype model of Huang et al. (2012, see Section 8.4), the experiments involve Multilingual Vectors by Faruqui and Dyer (2014), who learned vectors by first performing SVD on text in different languages, then applying canonical correlation analysis on pairs of vectors for words that align in parallel corpora. These vectors were trained on the WMT-2011 news corpus for English, French, German and Spanish.

The resulting representations were evaluated for their semantic and syntactic aspects in extrinsic sentiment analysis task, Word Similarity, Syntactic Relations by Mikolov, Synonym Selection (TOEFL), and phrase and sentence level Sentiment Analysis (Socher et al. 2013).

Mrkšić et al. (2016) present a counter-fitting method that injects both antonymy and synonymy constraints into vector space representations improving the vectors' capability for judging semantic similarity. The method gave new state-of-the-art performance on the SimLex-999 dataset and was demonstrated in the downstream task of dialogue state tracking (where the task is updating the system's distribution over user goals as the conversation progresses and new information becomes available), resulting in robust improvements across domains.

Word representations coalesce semantic similarity and conceptual association (Hill, Reichart, and Korhonen 2014). Furthermore, even methods that can distinguish similarity from association (e.g., based on syntactic co-occurrences) will generally fail to tell synonyms from antonyms (Mohammad, Dorr, and Hirst 2008). Distinguishing antonymy from similarity is critical for the dialogue state tracking task (DST), more specifically the restaurant domain, where systems should not recommend an "expensive pub in the south" when asked for a "cheap bar in the east". Counter-fitting, is a lightweight post-processing procedure in the spirit of the retrofitting introduced in the previous subsubsection.

Mrkšić et al. (2017) introduce Attract-Repel which jointly injects mono- and cross-lingual synonymy and antonymy in word embeddings, yielding semantically specialised[3] cross-lingual vector spaces. In practice, semantic transfer goes from high to lower-resource languages. Their evaluation obtains SOTA on SimLex semantic similarity datasets in six languages and in DST across multiple languages. Their multilingual DST models bring further performance improvements.

Mrkšić et al. term the retrofitting approach, i.e. when vectors are refined to satisfy constraints extracted from a lexicons such as WordNet, *semantic specialization*. Mrkšić et al. deploy the Attract-Repel algorithm in a multilingual setting, taking semantic relations from BabelNet and exploiting information from high-resource languages to improve the lower-resourced ones. They train their cross-lingual vector

3 They use British spelling, and we keep it, because this is a term.

spaces jointly, which brings benefits in the form of positive semantic transfer.

Mrkšić et al. demonstrate their efficacy both in intrinsic and downstream tasks. The former includes SOTA results on the four languages in the Multilingual SimLex-999 dataset and in lower-resource languages Hebrew and Croatian, where Mrkšić et al. collect evaluation datasets, and show that cross-lingual specialization significantly improves word vector quality.

Their downstream applications are motivated by improving the lexical coverage of supervised models. Mrkšić et al. consider again DST. Incorporating their specialised vectors into a SOTA neural network model for DST improves performance on English dialogues. In a multilingual spirit, Mrkšić et al. produce new Italian and German DST datasets, where Attract-Repel-specialised vectors leads to even stronger gains, and they train a single model that performs DST in all three languages, in each case outperforming the monolingual model.

The retrofitting models discussed so far specialize only the vectors of words from the constraints. Glavaš and Vulić (2018) use the external lexico-semantic relations to train an explicit retrofitting model (ExRf), which learns a global specialization function and specializes the vectors of words unobserved in the training data as well. They evaluate in intrinsic word similarity evaluation and two downstream tasks – lexical simplification and dialog state tracking. The authors also specialize vector spaces of new languages (i.e. unseen in the training) by coupling ExRf with shared multilingual distributional vector spaces.

The two prominent ways for external constraints are joint specialization models, which integrating the constraints into the distributional learning objective, and post-processing models, which fine-tune distributional vectors retroactively. In general, the latter outperform the former, and they can be applied to arbitrary distributional spaces but vectors of all unseen words remain intact.

Glavaš and Vulić propose explicit retrofitting (ExRf), which unifies the strengths of the two. ExRf is applicable to arbitrary embeddings, learns an explicit global specialization function, directly learns a specialization function in a supervised setting. It is implemented as a deep feedforward neural architecture. Glavaš and Vulić show that the proposed ExRf approach yields considerable gains in word similarity evaluation on standard benchmarks SimLex-999 (Hill+ 2015); SimVerb-3500 (Gerz+ 2016), and in two downstream tasks – lexical simplification and dialog state tracking. By coupling the ExRf model with shared multilingual embedding spaces, we can also specialize distributional spaces for unseen languages.

4.2.11 *Sub-word embeddins for rich morphology*

The next important step in the history of word embeddings is sub-word level modeling (Bojanowski et al. 2017), which we now discuss with an emphasis on rich morphology, keeping in mind that sub-word level modeling solves other kinds of out-of-vocabulary problems, like proper nouns, as well. For morphologically rich languages, word embeddings provide less consistent semantic representations due to higher variance in word forms. Moreover, these languages often allow for less constrained word order, which further increases variance. For the highly agglutinative Hungarian, semantic accuracy of word embeddings measured on word analogy tasks drops by 50-75% compared to English. In this section – which originally appeared as Döbrössy, Makrai, Tarján, and Szaszák (2019) – we describe experiments showing that embeddings learn morphosyntax quite well instead.

Therefore, we explore and evaluate several sub-word unit based embedding strategies – character n-grams, lemmatization provided by an NLP-pipeline, and segments obtained in unsupervised learning (`Morfessor`) – to boost semantic consistency in Hungarian word vectors. The effect of changing embedding dimension and context window size are also considered. Morphological analysis based lemmatization is found to be the best strategy to improve embeddings' semantic accuracy, whereas adding character n-grams is consistently counterproductive in this regard.

4.2.11.1 *Introduction*

Word embeddings show amazing capabilities in capturing and representing semantic relations within natural languages, which has also been demonstrated in analogical reasoning tasks (Mikolov, Yih, and Zweig 2013; Gladkova and Drozd 2016). They are also capable of learning morphosyntax, showing again a consistent mapping of grammatical operations, i.e. inflections (see Section 4.2.11.2). Word embeddings obtain such semantic and syntactic capabilities by matching the words to their observed contexts (or vice versa) when training an encoder-decoder network. Since the size of the word vector table is the vocabulary size times the embedding dimension, for languages with rich morphology (especially agglutinative ones), this results in huge matrices (Takala 2016). The vocabulary needs to be increased for morphologically rich languages to ensure a high enough coverage for the overall occurring words. To obtain a reliable estimate of word vectors, a larger training corpus is required so that theoretically the same convergence of the estimation can be reached than for a non agglutinative language. Furthermore, morphologically rich languages tend to express grammatical relations through suffixes (i.e. case endings) and hence let the word order become less constrained than in configurational languages. This can result in higher context variability, which translates again into less ac-

curate estimates (i.e. the effect of migrating words outside the context window can blur representations). Augmenting the size of the context window is not a effective counter-measure, as it will result again in higher variability of the context.

Bojanowski et al. (2017) proposes character level enhancement for word embeddings to overcome difficulties caused by unseen or rare words. It is demonstrated for a large set of languages that adding character n-grams to the embeddings can be a powerful way of generating word vectors for unseen words, and this improves both the semantic and the syntactic consistency (and accuracy) of the embeddings. However, Bojanowski et al. (2017) tests no highly agglutinative language for their embeddings' syntactic and semantic accuracy with and without n-grams.

We conduct proper evaluation on an analogy set for Hungarian (Makrai 2015) designed according to the standard Mikolov, Chen, et al. (2013), and show that the already weak baseline semantic accuracy consistently decreases when character n-grams are added. On the other hand, embeddings learn the complex Hungarian morphosyntax quite well.[4]

4.2.11.2 *Related work*

The closest work to ours is a concurrent study (Zhu, Vulić, and Korhonen 2019) of subword models especially for morphologically rich languages across different tasks. Unfortunately they miss Hungarian, which left a huge gap, as they find that performance is both language- and task-dependent. They find that unsupervised segmentation (e.g., BPE, Morfessor, see later in this section) is sometimes comparable to or even outperforms supervised word segmentation.

MORPHOLOGY IN WORD EMBEDDINGS. The morphologically informed approach to compositionally gained word embedding vectors start with Lazaridou et al. (2013) and Luong, Socher, and Manning (2013), who train a Recursive Neural Network, which builds representations for morphologically complex words from their morphemes.

The work of Soricut and Och (2015) can be regarded as the unsupervised counterpart of Mikolov, Yih, and Zweig (2013)-style analogical questions. Soricut induces morphological relations as the systematic difference of embedding vectors in an unsupervised manner. They evaluate on word-similarity.

4 This work was supported by the Hungarian National Research, Development and Innovation Office under contract ID FK-124413: 'Enhancement of deep learning based semantic representations with acoustic-prosodic features for automatic spoken document summarization and retrieval'. Márton Makrai was partially supported by project found 2018-1.2.1-NKP-00008: Exploring the Mathematical Foundations of Artificial Intelligence and National Research, Development and Innovation Office grant #120145.

Relying on existing morphological resources, Cotterell et al. (2016) introduce a latent-variable morphological model that extrapolates vectors for unseen words, and smoothes those of observed words over several languages.

Cao and Rei (2016) introduce a joint model for unsupervised segmentation and weighted character-level composition. Cotterell et al. (2018) compute supervised models for the same two sub-tasks of morphological analysis, also induces a canonical form (i.e. models orthographic changes).

Avraham and Goldberg (2017) argue that morphology-driven models confuse two aspects of the canonical form: their base form component is mostly responsible for *semantic* aspects of the similarity, while affixes, especially inflectional ones (e.g. *-s*), are mostly responsible for *morphological* similarity. They also investigate whether models behave differently on common and rare words. They conclude that a morphological component should be included only for tasks in which morphological similarity cannot be handled by other means.

LANGUAGE MODELING AND CHARACTERS. Morphologically compositional language modeling proper begins with Botha and Blunsom (2014)'s decoder in machine translation to morphologically rich languages, which is unsupervised with respect to morphological segmentation. Cotterell and Schütze (2015) augment the log-bilinear language model (LM, Mnih and G. Hinton (2007)) with a multi-task objective for morphological tags along with the next word.

Character n-gram features proved to be powerful as the basis of Facebook's fastText classifier (Joulin et al. 2016). Subword units based on byte-pair encoding have been found to be particularly useful for machine translation (Sennrich, Haddow, and Birch 2016), and even in models based on matrix factorization (Salle and Villavicencio 2018).

Ruder (2018) collects works augmenting embeddings with subword-level information for many applications in the form of a convolutional neural network or a BiLSTM.

A recent line of research (see Section 4.3.2 as well) aims at understanding the type of linguistic knowledge encoded in sentence and word embedding modules of neural machine translation (NMT) encoders and decoders or even in those of deep NLP models (Peters, Neumann, Iyyer, et al. 2018; N. A. Smith 2019), which have recently set new state-of-the-art results in many tasks. Belinkov et al. (2017a) found that character-based representations are much better for learning *morphology*, especially for low-frequency words; and lower layers of the encoder are better at capturing word structure, while higher layers focus more on word meaning. Representations learned from the NMT encoder turn out to be rich in morphological information, but those from the decoder are significantly poorer. This motivates Dalvi et al. (2017) to inject target morphology into the NMT decoder, which improves the translation

jelmondatával	→	jelmondat <poss> <cas<ins>>
akartak	→	akar <past> <plur>

Table 7: De-glutination (Nemeskey 2017)

quality. Dalvi et al. (2019) analyze individual neurons in *deep NLP models*. Their *linguistic correlation analysis task* investigate sensitivity for word-structure (morphology) among other linguistic properties.

HUNGARIAN. In their de-glutinatve method, Borbély, Kornai, et al. (2016) and Nemeskey (2017) split all inflectional prefixes (as well as some derivational ones, such as <compar>ative and <superlat>ive of adjectives) into separate tokens for better morphological generalization, see Table 7. Nemeskey opts for supervised morphological knowledge because of linguistic interpretability. Lévai and Kornai (2019) analyze Hungarian word embedding vectors grouped by the morphological tag of the corresponding word. They investigate whether the coherence of these classes correlate with the specificity or the frequency of the tag. Again, the readers interested in the most recent advances should consult papers like (Ács et al. 2021) and beyond.

4.2.11.3 *Experiments*

CORPUS, SEGMENTATION, AND EMBEDDINGS For training the word vector models, we rely on the fastText (Joulin et al. 2016) tool, which also allows for augmentation with character n-grams, if desired. We do not use stemming, but go instead for some more sophisticated analysis. As we explained, our primary goal is benchmarking the individual approaches.

For a true morphological analysis, we use the magyarlánc (Zsibrita, Vincze, and Farkas 2013) toolkit, which provides lemmatization in the form of a stem plus a suffix series, also decomposed into individual component morphemes. Although some disambiguation capability arises from sentence level part-of-speech tagging, magyarlánc may end up with several hypotheses for the morphological composition of the input word. Fortunately this happens rarely at the lemma level. If it does, the shortest lemma is used.

For unsupervised pseudo-morphemic analysis, we use Morfessor (Virpioja et al. 2013). Morfessor has been used to provide subword unit tokens for Automatic Speech Recognition in heavily agglutinative languages, with improved accuracy (Enarvi et al. 2017) over word based vocabularies and models. Morfessor is based on statistical machine learning. In order to reflect that the provided subword units are not true morphemes in the grammatical sense, they are called morphs.

The text corpus we use is a contemporary dump of Hungarian language web pages constructed for this paper, which covers mostly online

Parameter	Value range
Frequency cut-off	5
Min length of char ngram	none or 3
Max length of char ngram	none or 6
Embedding dimension	100-200
Context window	5–25
Learning rate (α)	0.05
α update interval	100
Number of epochs	15
Negative sampling loss	yes
Negative samples	5
Pretraining	none

Table 8: Embedding vector trainer parameters.

newspapers in various fields from years 2014-2018. The corpus has over 70 M word tokens. Text normalization is performed with a Python script.

ANALOGICAL QUESTIONS Our approach is to train word embeddings in different scenarios and assess syntactic and semantic accuracy based on a Hungarian analogy test (Makrai 2015) that has been constructed according to (Mikolov, Chen, et al. 2013). For the semantic accuracy, we use `country-capital` and `country-currency` pairs. For the syntactic accuracy, we use `singular-plural` for nouns, `present-past tense` for verbs, and `base vs comparative forms` for adjectives.

FASTTEXT SETTINGS There are three main parameters which are controlled during the experiments: (i) whether we use character n-gram augmentation or not; (ii) the size of the context window; and (iii) the target dimension of the resulting embedding vectors. We preferred to preserve all other parameters of fastText at their default value. The most important of these parameters are summarized in Table 8.

EMBEDDING STRATEGIES

WORD VECTORS (W). This constitutes our baseline. A standard word embedding is trained with fastText, no prior stop word filtering is applied.

LEMMA VECTORS (L). The magyarlánc toolkit is used for morphological analysis. Lemmas are identified and used as embedded entities. Note that whereas ambiguity on the entire morphological composition

may arise, ambiguity affecting the lemma's surface form is rare. If this still occurs, the shortest form is used.

MORF VECTORS (M). Running Morfessor yields a morph based split-up. Morfs become the modeling unit (subword unit). As an alternative, using the **root (R)** yielded by Morfessor is evaluated as well. The word embedding is trained on the corpus with words divided into segments (as if they were separate words). During testing in analogical questions, query words are also spitted to segments, and their vectors are computed as the sum of the segments' vectors.

VECTOR DIMENSION is changed between 100 and 200. We did not consider using higher dimensions to avoid making downstream applications heavy.

We will refer to the individual setups by specifying the unit out of {W, L, M, R} and the dimension, e.g. L200 will refer to lemma as unit and 200-dimensional embeddings.

4.2.11.4 *Results*

EXTENDING THE CONTEXT WINDOW As we pointed out in Section 4.2.11.1, using wider context may help in overcoming the difficulties resulting from the less constrained word order of Hungarian. A wider context window allows for capturing words further apart, but it may have an adverse effect as well, because the context becomes more noisy (variable). Relative data sparsity may also be a problem when a larger context is considered. So basically our research question related to the context of a word is that whether the benefits of capturing further apart words can be superior compared to the negative effect of increasing variance w.r.t the occurring context words.

It has been reported (Lebret and Collobert 2015) that semantic analogical questions benefit from larger windows, while syntactic ones do not. On the contrary, experimenting with SVD models and different window sizes, Gladkova and Drozd (2016) find that all categories of analogical questions are best detected between window sizes 2–4, although a handful of them yield equally good performance in larger windows. They find no one-on-one correspondence between semantics and larger windows. We consider unusually large contexts of up to 25 words (see Table 8).

Semantic and syntactic accuracy with 100 dimensional embeddings are shown in Figures 8 and 9, respectively. Comparing strategies, using the lemma (L) for embedding is yielding the highest semantic accuracy. Regarding the context window, our hypothesis that long context windows may be a better fit is confirmed. All the four strategies consistently show increasing semantic accuracy as context window is extended to cover 21 units. Compared to W, L embeddings yield higher semantic

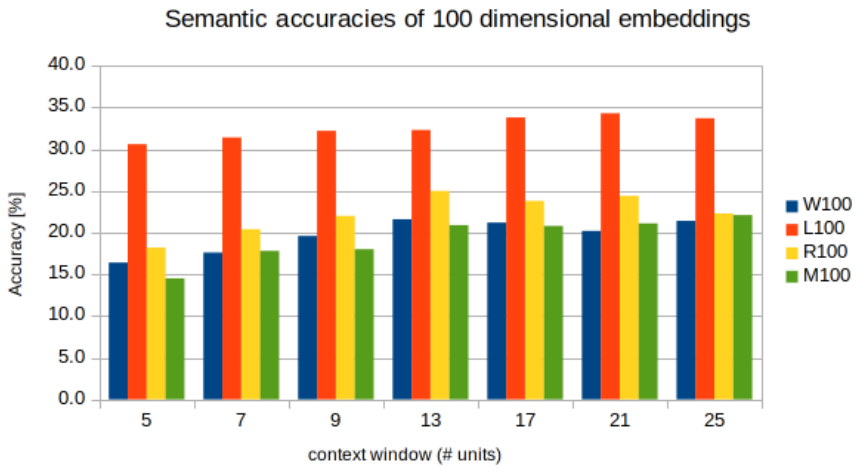

Figure 8: Semantic accuracy of Hungarian 100 dimensional embeddings with different strategies.

accuracy by 75%. Nevertheless, syntactic accuracy tends to decrease when extending the context window, which is a negative effect, most likely resulting from the higher variation seen in a larger window.

ADDING CHARACTER n-GRAMS In contrast to many other languages (Bojanowski, Joulin, and Mikolov 2016), the highly agglutinative Hungarian cannot profit from adding character n-grams to the embeddings: semantic (but also syntactic) accuracy gets lower. We suppose that this happens because agglutination is frequent and hence word vectors become universal (i.e. they cannot specialize for the context). The less constrained word order plays a role in this, too.

Figure 10 shows how semantic and syntactic accuracy change when adding character n-grams (sem+chr and syn+chr, respectively) in the W100 case. We present again a trend with increasing context window size on the horizontal axis to allow for easy comparison with the previous results.

Regarding semantic accuracy, no benefit is registered when adding character n-grams with any of the 4 investigated embedding strategies.

Adding character n-grams becomes helpful at the syntax level in some cases, syntactic accuracy augments for the L100, L200 and R200 scenarios. Nevertheless, the basis is very low as for using the lemmas or Morfessor roots, most of the morphosyntactic information is lost. Not surprisingly, semantics improves with a large window, while morphosyntax does not.

EMBEDDING DIMENSION Figure 11 compares the semantic accuracy of 100 and 200 dimensional scenarios with a context window of 21. Increasing the embedding dimension has a positive effect on semantic accuracy, as far as up to 50% relative increase in accuracy. Accuracy

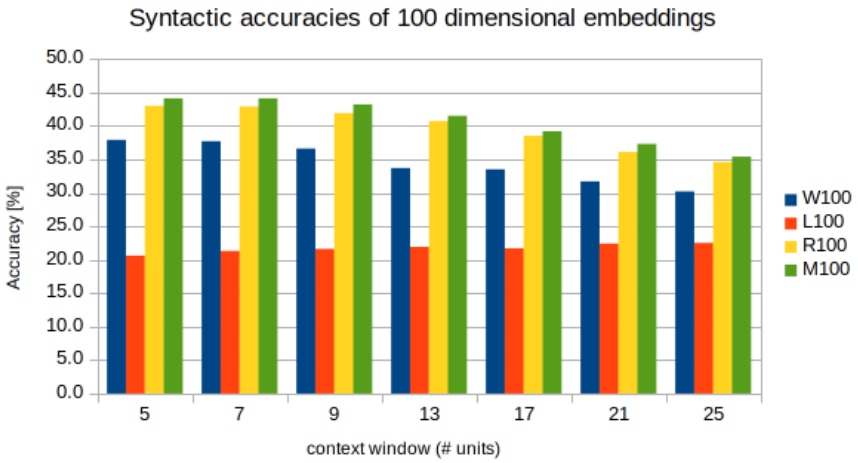

Figure 9: Syntactic accuracy of Hungarian 100 dimensional embeddings with different strategies.

capital-common-countries	66.0% (101/153)
capital-world	40.3% (2595/6441)
county-center	18.2% (12/66)
currency	6.4% (26/406)
family	16.5% (15/91)
Semantic	38.41% (2749/7157)

Table 9: Results in individual semantic relations with the best setting (magyarlánc, window 21, dimension 200, no character n-grams).

in individual relations (whose importance has been shown by Gladkova and Drozd (2016)) are reported in Table 9. We can again observe that adding character *n*-grams consistently results in decreased semantic accuracy.

Increasing embedding dimensions above 200 could be expected to yield further improvement is semantic accuracy, but we did not address this issue in our current work, which focuses mostly on the modeling unit and its optimal context.

4.2.11.5 *Future work*

Future work may investigate whether results generalize to other embedding algorithms (besides fastText, the original and the enhanced (Mikolov et al. 2018) `word2vec` and the GloVe (Řehůřek and Sojka 2010) implementations of the *continuous bag of words* and the *skip-gram* models could be tried); extend the ablation over dimensionality up to a few hundred dimensions; and analyze other morphologically rich languages (e.g. Finnish, Turkish, or Slavic languages). The bottleneck is that we

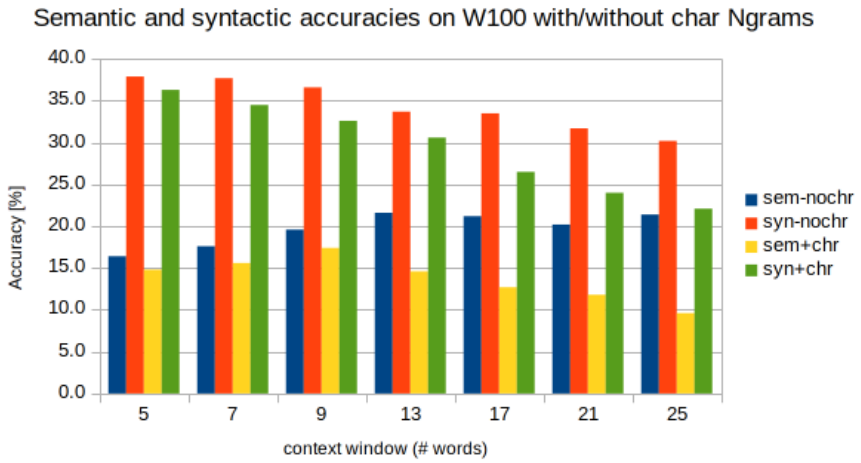

Figure 10: Semantic and syntactic accuracy of Hungarian 100 dimensional word embeddings with (chr) and without (nochr) character n-grams.

are restricted to languages to which the analogical questions have been translated. As a reviewer noted, the semantic part of the Mikolov-style analogical questions consist of a handful of semantic relations between named entities. It is questionable how appropriate it is to use them for the evaluation of the embedding strategies, especially that of encoding lexical semantic relations and not the world knowledge. Gladkova and Drozd (2016) examine Mikolov, Yih, and Zweig (2013)-style analogical questions systematically, finding that different systems shine at different sub-categories of the morphological and semantic tasks. They publish a test set which is more difficult than existing ones. Translating this test set to morphologically rich languages would be very useful.

4.2.12 *The offset is naked*

The basic way of evaluating static word embeddings have been intrinsic evaluations, namely similarities and analogies. Both methods have serious shortcomings – we illustrated this for similarities in Section 4.2.9. Now we turn to a critical reflection on what have been called the vector offset method, relational similarity, or word analogies.

Levy et al. (2015) argue that supervised methods of hypernymy are memorizing whether the hypernym candidate is a "prototypical hypernym", i.e. a category, irrespective of the word to be categorized. They compare four compositions for representing (x, y) (e.g. $x = \text{cat}, y = \text{animal}$) as a feature vector: besides the standard concatenation $x \oplus y$ and difference $y - x$, they use the diagnostic "only x", and "only y". The finding is that models just learn whether y is a likely "category" word – a prototypical hypernym – and, to a lesser extent, whether x is a likely "instance" word. This extends to other inference relations,

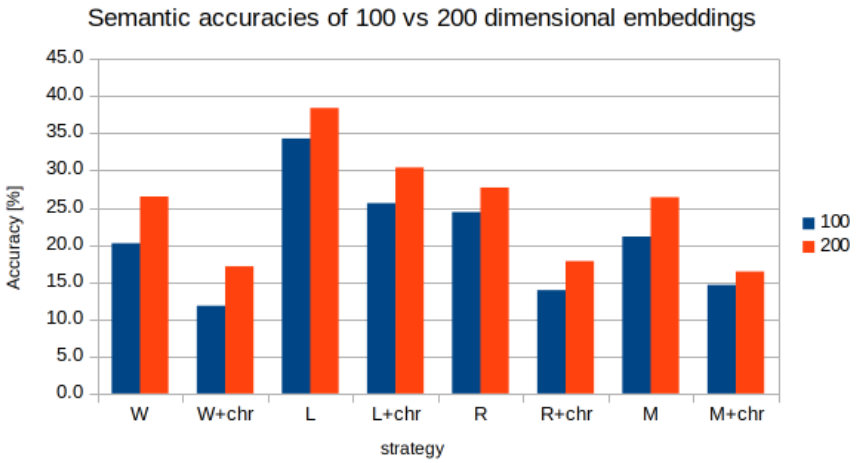

Figure 11: Semantic accuracy of Hungarian 100 and 200 dimensional embeddings with different strategies; context window covers 21 units.

such as meronymy. To test the hypothesis, the authors manipulate the test pairs by inserting mismatched pairs, e.g. *(banana, animal)*.

The word embeddings they use include interpretable PPMI-based ones, which enable them to look for prototypical hypernym contexts. Besides dataset-specific contexts like *psychosomatic* `-1` (*word* ± `i` denotes the context where the *i*th word to to right/left if *word*), they find domain-independent indicators of category, e.g. *any* `-1`, *every* `-1`, and *kinds* `-2`, and even relics of the Hearst patterns in all datasets: *other* `-1`, *such* `+1`, *including* `+1`, etc., and their analogons e.g. *such* `-2`.

Linzen (2016) notes that in analogical tasks

$$x = a^* - a + b,$$

if a^* and a are very similar to each other (as *scream* and *screaming* are likely to be) the nearest word to x may simply be the nearest neighbor of b. If in a given set of analogies the nearest neighbor of b tends to be b^*, the answer may be correct regardless of the consistency of the offsets. He proposes new baselines that perform the task without using the offset $a^* - a$, and measures how the performance is affected by *reversing* the direction of each analogy problem (which should not affect its accuracy).

4.2.12.1 *Theoretical critique of vector analogy*

Rogers, Drozd, and Li (2017) criticize the vector analogy method on theoretical grounds. Given the vital role that analogical reasoning plays *in human cognition*, automated analogical reasoning could become a game-changer in many fields. The method is already used in many downstream NLP tasks, such as splitting compounds, semantic search, and cross-language relational search. One way to explain the current limitations is to attribute them to the imperfections of the models

and/or the corpora. With this view, in a perfect VSM, any linguistic relation should work. The alternative explored by Rogers, Drozd, and Li is that there are both theoretical and mathematical issues with analogical reasoning with word vectors and 3CosAdd (see Section 4.2.7).

In the authors' view, the most fundamental term is not analogy, but *relational similarity*, i.e. that pairs of words may hold similar relations. We speak of similarity rather than identity: instances of a single relation may still have significant variability in how characteristic they are of that class.

"Classical" analogical reasoning follows roughly this template: objects X and Y share properties a, b, and c; therefore, they may also share the property d. For example, both Earth and Mars orbit the Sun, have at least one moon, revolve on axis, and are subject to gravity; therefore, if Earth supports life, so could Mars. The NLP move from relational similarity to analogy follows the use of the term by Turney.

Analogy was once rejected in generative linguistics as a mechanism for language acquisition through discovery, although now it is making a comeback. It has been criticized for ambiguity, guesswork and puzzle-like nature.

The paper has been referred to as *Mikolov cheated!*, because they point out that 3CosAdd, as initially formulated by Mikolov, Yih, and Zweig (2013), "dishonestly" excludes a, a_o and b from among potential b_os.

The authors present a series of experiments performed with the BATS dataset, which has more relations and is more difficult than the original Google test. BATS is balanced across derivational and inflectional morphology, lexicographic and encyclopedic semantics (10 relations of each type). They explain lower performance on *derivational morphology questions* as opposed to inflectional or encyclopedic semantics: *man* and *woman* are reasonably similar distributionally, as they combine with many of the same verbs: both men and women sit and sleep, but the same could not be said of words derived with prefixes that change POS.

Another, purely logical problem is exemplified by *snow: white :: sugar: ?white*, where, in the dishonest setting, the correct answer is a priori excluded. In BATS data, this factor affects several semantic categories, including country:language, thing:color, animal:young, and animal:shelter.

Rogers, Drozd, and Li hypothesize that the more *crowded a particular region* is, the more difficult it should be to hit a particular target. Estimating density as the similarity to the 5th neighbor, they get the counter-intuitive results that denser neighborhoods actually yield higher scores.

They consider LRCos, a method based on *supervised* learning from a set of word pairs. The model learns a representation of the target class with a supervised classifier. The question is this: what word is

the closest to king, but belongs to the "women" class? The accuracy of LRCos is much higher than the top-1 3CosAdd or 3CosMul, and its "honest" version performs just as well as the "dishonest" one.

4.2.13 *Frequency effects in cosine similarity*

Faruqui et al. (2016) review the main problems with word similarity evaluations, and they discuss frequency effects in cosine similarity (besides the subjectivity of the task; the confusion of semantic and task-specific similarity; the lack of standardized splits and overfitting; the low correlation with extrinsic evaluation, i.e. with tasks like text classification, parsing, and sentiment analysis; and the absence of statistical significance).

Vectors of frequent words are longer as they are updated more often during training (Turian, Ratinov, and Bengio 2010). In Faruqui et al.'s view, ideally the relatively small number of frequent words should be evenly distributed through the space, while rare words should cluster around related, but more frequent words.

However, vector-spaces contain hubs, i.e. vectors that are close to a large number of other vectors in the space. In word vector-spaces, this takes the form of words that have high cosine similarity with a large number of other words (Dinu, Lazaridou, and Baroni 2015), as we will discuss in Sections 8.1.3 and 8.5.1. Schnabel et al. (2015) further refine this hubness problem to show a power-law relationship between the frequency-rank r of a word (i.e. the rank of a word in vocabulary of the corpus sorted in decreasing order of frequency.) and the frequency-rank of its neighbors: the average rank a of the 1000 nearest neighbors of a word follows: $a \approx 1000r^{0.17}$.

The last problem Faruqui et al. discuss is related to the main problem with word embeddings of the type investigated in this section: the inability to account for polysemy. As we will see in Section 8.2, there has been progress on obtaining multiple vectors per word-type to account for different word-senses, but the practical advantage of word embeddings with more but fixed vectors to account for different senses remained modest (Li and Jurafsky 2015), and the real solution was contextualized word representations provided by deep language models, which arguably brought a new paradigm in NLP, to which will now turn.

4.3 ATTENTION AND DEEP LANGUAGE MODELS

The contributions of this thesis are based on static word embeddings, i.e. the kind discussed so far, but we would like to put our investigation in the context of the advances of the past few years. Deep neural networks have defined the state-of-the-art in many areas of NLP.

Deep neural networks and *deep learning* mean machine learning of a model that consists of layers from the input layer through hidden layers to the output layer, and calculates higher and higher level features. Deep learning first brought breakthroughs in speech technology (Dahl et al. 2011) and computer vision (Krizhevsky and Sutskever 2012). The ImageNet moment of NLP, as Ruder (2018) called it, arrived in 2018.

> Pretraining entire models to learn both low and high level features has been practiced for years by the computer vision (CV) community. Most often, this is done by learning to classify images on the large ImageNet dataset. ULMFiT, ELMo, and the OpenAI transformer have now brought the NLP community close to having an "ImageNet for language"—that is, a task that enables models to learn higher-level nuances of language, similarly to how ImageNet has enabled training of CV models that learn general-purpose features of images. (https://ruder.io/nlp-imagenet/)

Up to this point of the thesis, we have been chronological and didactic. Main contribution chapters will be similar, even self-contained in many cases. This section provides, however, just some flashes for the reader somewhat familiar with deep learning of language. Those with less background in machine learning may skip to the first foreground chapter, Chapter 5. Where citations are omitted, they can be found in the corresponding paper we just summarize.

4.3.1 *Deep pretrained models for NLP*

Qiu et al. (2020, Section 2.4.2) summarize the history of pretrained deep NLP models as follows: McCann et al. (2017) pre-trained a deep *LSTM encoder from an attentional sequence-to-sequence model* with machine translation objective, and used the context vectors (CoVe) output by the pretrained encoder. Peters, Neumann, Iyyer, et al. (2018) pre-trained 2-layer LSTM encoder with a *bidirectional language model (BiLM)*, consisting of a forward LM and a backward LM. Contextual representations output by the pre-trained BiLM, ELMo (Embeddings from Language Models) brought large improvements on a broad range of tasks. Flair (Akbik, Blythe, and Vollgraf 2018) captured word meaning with contextual string embeddings pre-trained with *character-level* LM. Ramachandran, Liu, and Le (2017) significantly improved the seq2seq models by unsupervised pre-training. The weights of both the encoder and the decoder are initialized with pretrained weights of two language models and then fine-tuned with labeled data.

ULMFiT (Universal Language Model Finetuning, Howard and Ruder (2018)) fine-tuned a *pre-trained LM* for text classification, achieving state-of-the-art results on six widely-used text classification datasets.

ULMFiT training consists of 3 phases: pre-training LM on general-domain data; fine-tuning LM on target data; and fine-tuning on the target task. Their *fine-tuning* strategies include discriminative fine-tuning, slanted triangular learning rates, and gradual unfreezing. Since ULMFiT, fine-tuning has become the mainstream approach to adapt PTMs for the downstream tasks.

Very deep PTMs have shown their powerful ability in learning universal representations, including OpenAI GPT (Generative Pre-training, Radford et al. (2018)) and BERT (Bidirectional Encoder Representation from Transformer, Devlin et al. (2018)). Besides LM, an increasing number of self-supervised tasks are proposed to make the PTMs capturing more knowledge form large scale text.

4.3.2 *BERTology*

Transformer-based models are now widely used in NLP, and much work has been done to understand their inner workings. The stream of papers seems to be accelerating rather than slowing down. Here we summarize the findings of Rogers, Kovaleva, and Rumshisky (2020), who synthesize over 40 analysis studies, overview the proposed modifications and the training regime, and offer directions for further research.

4.3.2.1 *Introduction*

Transformers (Vaswani et al. 2017) took NLP by storm, offering enhanced parallelization and better modeling of long-range dependencies. The best model is BERT (Devlin et al. 2019) which obtained state-of-the-art results in many benchmarks, and it has been integrated in Google search, improving an estimated 10% of queries. However, this family of models has little cognitive motivation, and the size of these models limits their training and study. Rogers, Kovaleva, and Rumshisky focus on the papers investigating the types of knowledge learned by BERT, where this knowledge is represented, how it is learned, and the methods proposed to improve it.

4.3.2.2 *Overview of BERT architecture*

BERT is a stack of Transformer encoder layers with multiple *heads*, i.e. fully-connected neural networks augmented with a self-attention mechanism. For every input token in a sequence, each head computes key, value and query vectors, which are used to create a weighted representation. The outputs of all heads in the same layer are combined and run through a fully-connected layer. Each layer is wrapped with a skip connection and layer normalization

The conventional workflow is pre-training and fine-tuning. Pretraining uses two semi-supervised tasks: masked language modeling (MLM, prediction of randomly masked input token), and next sentence predic-

tion (NSP, predicting if two input sentences are adjacent to each other). In fine-tuning for downstream applications, one or more fully-connected layers are typically added on top of the final encoder layer.

The representations are computed as follows: the model tokenizes the given word into wordpieces, and then combines three embedding layers (token, position, and segment). The special token [CLS] is used for classification predictions, and [SEP] separates input segments. Two sizes fit all: base and large, varying in the number of layers, their hidden size, and number of attention heads.

4.3.2.3 *What knowledge does BERT have?*

Analysis approaches include fill-in-the-gap probes of BERT's MLM, that of self-attention weights, and probing classifiers using different BERT representations as inputs.

SYNTACTIC KNOWLEDGE Representations are hierarchical rather than linear. There is something akin to syntactic tree structure in addition to the word order information. BERT has information about parts of speech, syntactic chunks and roles. Knowledge of syntax is partial, not enough to recover the labels of distant parent nodes in the syntactic tree. The syntactic structure is not directly encoded in self-attention weights, but they can be transformed to reflect it. Dependency trees have been extracted directly from self-attention weights but without quantitative evaluation. Transformation matrices recover much of the Stanford Dependencies formalism for PennTreebank data.

BERT representations have been approximated with Tensor Product Decomposition Networks, concluding that dependency trees are the best match among 5 decomposition schemes, but differences are very small. BERT takes subject-predicate agreement into account in the cloze task even with distractor clauses and meaningless sentences. BERT is able to detect the presence of negative polarity items (e.g. "ever") and the words that allow their use (e.g. "whether") but not scope violations. BERT does not understand negation, and it is insensitive to malformed input: predictions were not altered even with shuffled word order, truncated sentences, or removed subjects and objects. Models are disturbed by nonsensical input (adversarial attacks).

SEMANTIC KNOWLEDGE Fewer studies were devoted to BERT's knowledge of semantics. Entity types, relations, semantic roles, and proto-roles have been detected with probing classifiers. BERT has some knowledge for *semantic roles*. Ettinger (2020) shows with an MLM probing study that the model prefers incorrect fillers for semantic roles that are semantically related to the correct ones, to those that are unrelated e.g. *to tip a chef* to "to tip a robin", see Section 4.3.5.

BERT struggles with representations of *numbers* (addition, number decoding, floating point numbers). The problem may be with word-

piece tokenization: numbers of similar values can be divided up into substantially different word chunks.

BERT is surprisingly brittle to *named entity* replacements: replacing names in the coreference task changes 85% of predictions. This suggests that the model does not form a generic idea of named entities, although its F1 scores on NER probing tasks are high. Fine-tuning BERT on Wikipedia entity linking "teaches" it additional entity knowledge, which suggests that it did not absorb all the relevant entity information during pre-training on Wikipedia.

WORLD KNOWLEDGE MLM has been adapted for knowledge induction by filling in the blanks, e.g. "Cats like to chase [ ]". Besides a probing study of world knowledge in BERT, evidence comes from many practitioners using BERT to extract knowledge. For some relation types, vanilla BERT is competitive with knowledge base methods. BERT generalizes well to unseen data, but we need good template sentences. There has been research on the automatic extraction and augmentation of such templates.

BERT cannot reason based on its world knowledge. It can *guess* the affordances and properties of many objects, but it has no information about their interactions. E.g. it knows that people can walk into houses, and that houses are big, but it cannot infer that houses are bigger than people. Its performance drops with the number of necessary inference steps. Some of BERT's success in factoid knowledge retrieval comes from learning stereotypical character combinations, e.g. that a person with an Italian-sounding name is Italian.

LIMITATIONS Some researchers remark that "the fact that a linguistic pattern is not observed by our probing classifier does not guarantee that it is not there, and the observation of a pattern does not tell us how it is used." A hot question is how complex a probe should be: If a more complex probe recovers more information, to what extent are we still relying on the original model? Different probing methods may lead to complementary or even contradictory results. A given method might also favor one model over another. E.g., RoBERTa trails BERT with one tree extraction method, but leads with another. The choice of linguistic formalism also matters.

We should focus on identifying what BERT actually relies on at inference time. Amnesic probing aims to specifically remove certain information, and see how it changes performance. This method has shown that e.g. language modeling does rely on part-of-speech information.

Information-theoretic probing approaches include estimating the mutual information between the learned representation and a given linguistic property. Some researchers quantify the amount of effort needed to extract some information, which is more important than the amount of information in the representation. The mathematical formalism is

minimum description length needed to communicate both the probe size and the amount of data required for it to do well on a task.

4.3.2.4 *Localizing linguistic knowledge*

BERT EMBEDDINGS In studies of BERT, the term *embedding* refers to the output of a Transformer layer (typically, the final one). Every token contains at least some information about the *context*. Both conventional static embeddings and BERT-style embeddings can be viewed in terms of mutual information maximization.

Distilled contextualized embeddings better encode lexical semantic information, i.e. they are better at traditional word-level tasks such as word similarity. The methods to distill a contextualized representation into a static one include aggregating the information across multiple contexts, encoding "semantically bleached" sentences that rely almost exclusively on the meaning of a given word (e.g. *This is* $\langle\ \rangle$), and using contextualized embeddings to train static embeddings. Distillation to static embedding is useful because interpretability methods for static embeddings are more diverse and mature than those available for their dynamic counterparts.

It has been studied how similar the embeddings for identical words are in every layer, reporting that later BERT layers are more context-specific. In the earlier Transformer layers, MLM forces the acquisition of contextual information at the expense of the token identity, which gets recreated in later layers. To what extent do models capture phenomena like polysemy and homonymy? BERT embeddings form distinct clusters corresponding to word senses. The model is successful at word sense disambiguation. Representations of the same word depend on the position of the sentence in which it occurs, likely due to the NSP objective, what is desirable from the linguistic point of view, and could be a promising avenue for future work.

The standard way to generate *sentence* or *text* representations for classification is to use the [CLS] token, the concatenation of token representations, or the normalized mean.

SELF-ATTENTION HEADS Several classifications of attention heads have been proposed in different studies:

- attending to the word itself, to previous/next words and to the end of the sentence,
- attending to previous/next tokens, [CLS], [SEP], punctuation, and attending broadly over the sequence, or
- the 5 attention types in Figure 12: Vertical, Diagonal, Vertical + diagonal, Block, and Heterogeneous.

Heads with linguistic functions

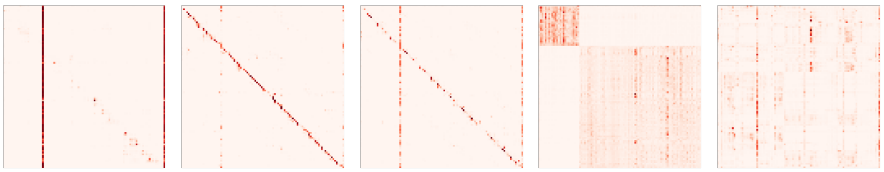

Figure 12: Typical self-attention patterns (Kovaleva et al. 2019). Both axes on every image represent BERT tokens of an input example, and colors denote absolute attention weights (darker colors stand for greater weights). The first three types are most likely associated with language model pre-training, while the last two potentially encode semantic and syntactic information.

The "heterogeneous" attention pattern could be linguistically interpretable, and a number of studies focused on identifying the functions of the heads.

There are BERT heads that attended significantly more than a random baseline to words in certain *syntactic* positions. Datasets and methods used in these studies differ, but there is some consistency that some heads attend to words in `obj` role more than the positional baseline. Evidence for `nsubj`, `advmod`, and `amod` varies between studies. The overall conclusion is also supported by a study in machine translation context. Even complex dependencies like `dobj` may be encoded by a combination of heads, but the corresponding work is limited to qualitative analysis.

No single head has the complete syntactic tree information, but a BERT head can directly be used for coreference classification on par with a rule-based system, what is remarkable because coreference classification requires quite a lot of syntactic knowledge. Attention weights are weak indicators of subject-verb agreement and reflexive anaphora. Instead of serving as strong pointers between related tokens, they were close to a uniform attention baseline, but there was some sensitivity to different types of distractors coherent with psycholinguistic data, see Section 4.3.5.

Morphological information in BERT heads has not been addressed, but with a sparse attention variant in the base Transformer, some attention heads appear to merge BPE-tokenized words. Semantic relations, core frame-semantic relations, as well as lexicographic and commonsense relations have been studied, but a head ablation study showed that heads related to some of these problems were not essential for BERT's success on GLUE tasks.

The popularity of self-attention as interpretation is due to the idea that "attention weights have a clear meaning: how much a particular word will be weighted when computing the next representation for the current word." This has been much debated. In a multi-layer model where attention is followed by a non-linear transformation, the patterns in individual heads do not provide a full picture. Many current papers are accompanied by attention visualizations, and visualization tools,

but analysis is mostly qualitative, often with cherry-picked examples, and should not be interpreted as evidence.

Attention to special tokens

Most self-attention heads do not directly encode any nontrivial linguistic knowledge; at least after fine-tuning on GLUE, less than 50% of heads exhibit the "heterogeneous" pattern. Much of the heads have the vertical pattern (attnding to [CLS], [SEP], and punctuation), what is likely related to the overparametrization issue. Norms of attention-weighted input vectors yield a more intuitive interpretation of self-attention reducing the attention to special tokens, but it is still not the case that most heads that do the "heavy lifting" are even potentially interpretable. Some work focuses on inter-word attention and simply excludes special tokens, which is a questionable method, as attention to special tokens actually matters at inference time.

The functions of special tokens are not yet well understood. [CLS] is typically viewed as an aggregated sentence-level representation – although all token representations also contain at least some sentence-level information. Some researchers experiment with encoding Wikipedia paragraphs with base BERT to consider specifically the attention to special tokens, noting that heads in early layers attend more to [CLS], in middle layers to [SEP], and in final layers to periods and commas. The function of attending to special tokens might be a kind of "no-op": a signal to ignore the head if its pattern is not applicable to the current case. While attention to special tokens increases, their importance for prediction drops. After fine-tuning, both [SEP] and [CLS] get a lot of attention, depending on the task.

BERT LAYERS BERT's input is a combination of token, segment, and positional embeddings. Lower layers have the most *linear word order* information. Knowledge of linear word order decreases around layer 4 (i.e. the middle), and that of *hierarchical sentence structure* increases, as detected by the probing tasks of predicting the index of a token, the main auxiliary verb, and the sentence subject.

There is consensus among studies with different tasks, datasets and methodologies that *syntactic information* (syntactic tree depth, subject-verb agreement, and syntactic probing) is the most prominent in the middle BERT layers. This must be related to that the middle layers of Transformers are overall the best-performing and the most *transferable* across tasks. There is conflicting evidence about syntactic *chunks*: Some researchers draw parallels to the order of components in a typical NLP pipeline from POS-tagging to dependency parsing to semantic role labeling; others show that lower layers were more useful for chunking, while middle layers were more useful for parsing; yet others find the opposite: both POS-tagging and chunking were performed best at the middle layers, in both BERT-base and BERT-large.

The *final layers* of BERT are the most task-specific: In pre-training, this means specificity to the MLM task, which would explain why the middle layers are more transferable. In fine-tuning, it explains why the final layers change the most.

Semantics is spread across the entire model. While most of syntactic information can be localized in a few layers, in semantic tasks, certain non-trivial examples get solved incorrectly at first but correctly at higher layers, e.g. predicate-argument relations help to disambiguate part-of-speech. This is rather to be expected: semantics permeates all language, and linguists like Goldberg (2006) debate whether meaningless structures can exist at all. What does stacking much more Transformer layers actually achieve in BERT in terms of the spread of semantic knowledge, and is that beneficial? Base and large BERTs shows the same overall pattern of cumulative score gains, only more spread out in the large BERT. This picture is disputed by other researchers, who place "surface features in lower layers, syntactic features in middle layers and semantic features in higher layers", but only one SentEval semantic task in the corresponding study actually topped at the last layer, three others peaked around the middle and then degraded by the final layers.

4.3.2.5 *Training BERT*

MODEL ARCHITECTURE CHOICES The most systematic study of BERT's architecture investigated the number of layers, heads, and model parameters, varying one option and freezing the others. The number of heads was not as significant as the *number of layers*, consistently with research that found the middle layers to be the most transferable. Larger hidden representation size was consistently better, but the gains varied by setting.

IMPROVEMENTS TO THE TRAINING REGIME Regarding the batch size, large-batch training (8k examples) improves both the language model perplexity and downstream task performance. With a batch size of 32k, BERT's training time can be significantly reduced with no degradation in performance.

Embedding values of the trained [CLS] token are not centered around zero, its normalization stabilizes the training, resulting in a slight performance gain on text classification tasks. "Warm-start", i.e. training in a recursive manner, where the shallower version is trained first and then the trained parameters are copied to deeper layers, achieves 25% faster training speed with similar accuracy to the original BERT on GLUE tasks.

PRE-TRAINING BERT The original BERT is a bidirectional Transformer pre-trained on two tasks: next sentence prediction (NSP) and masked language model (MLM). Pre-training is the most expensive part

of training BERT, and it would be informative to know how much benefit it provides. On some tasks, a randomly initialized and fine-tuned BERT obtains competitive or higher results than the pre-trained BERT. Most weights of pre-trained BERT are useful in fine-tuning, although there are "better" and "worse" subnetworks. One explanation is that pre-trained weights help the fine-tuned BERT find wider and flatter areas with smaller generalization error, which makes the model more robust to overfitting. Most new models' gains are often marginal, and estimates of model stability and significance testing are very rare.

The following topics have been investigated to improve pre-training.

How to mask

There are systematic experiments with corruption rate and corrupted span length; diverse masks for training examples within an epoch; masking every token in a sequence instead of a random selection; replacing the MASK token with [UNK] token, to help the model learn a representation for unknowns that could be useful for translation; and maximizing the amount of information available to the model by conditioning on both masked and unmasked tokens, and letting the model see how many tokens are missing.

What to mask

Alternatives include full words instead of word-pieces and spans rather than single tokens (predicting how many are missing). Masking phrases and named entities improves representation of structured knowledge.

Alternatives to masking

Experiments have been performed for replacing and dropping spans; deletion, infilling, sentence permutation and document rotation; for predicting whether a token is capitalized and whether it occurs in other segments of the same document; training on different permutations of word order in the input with the objective of maximizing the probability of the original word order; and the detection of tokens that were replaced by a generator network.

NSP alternatives and additional tasks

Removing NSP does not hurt or slightly improves performance. It has been replaced with the task of predicting both the next and the previous sentences; or identifying swapped sentences. Another model includes sentence reordering and sentence distance prediction with two new tasks on two levels. On the token-level: it has to be predicted whether a token is capitalized and whether it occurs in other segments of the same document; and the segment-level tasks include sentence reordering, sentence distance prediction, and supervised discourse relation classification. In another approach, both NSP and token position embeddings have been replaced by a combination of paragraph, sentence, and token index embeddings. Utterance order prediction for multiparty dialogue has also been proposed. Rogers, Kovaleva, and Rumshisky cites cross-lingual work as well.

Approaches include combining MLM with some other tasks: simultaneous learning of 7 tasks, including discourse relation classification and predicting whether a segment is relevant for IR; latent knowledge retrieval; knowledge base completion. Continual learning means sequential pre-training on a large number of tasks, each with their own loss which are then combined.

Pretraining data

Several studies explored the benefits of increasing the corpus volume; longer training; explicit linguistic information, both syntactic and semantic; using the label for a given sequence from an annotated task dataset (e.g. sentiment analysis); and learning representations for rare words separately.

The idea of explicitly supplying structured knowledge has been experimented with in different ways, including entity-enhanced models (including entity embeddings as input or adapting entity vectors to BERT representations); an additional pre-training objective of knowledge base completion; modifying the standard MLM task to mask named entities; training with MLM objective over text + linearized table data; or enhancing RoBERTa with both linguistic and factual knowledge with task-specific adapters.

FINE-TUNING BERT The *pre-training + fine-tuning* workflow is a crucial part of BERT. Pre-training is supposed to provide task-independent linguistic knowledge, while the fine-tuning process would presumably teach the model to extract information from the representation.

During fine-tuning BERT, the most changes for 3 epochs occurred in the last two layers. Those changes caused self-attention to focus on [SEP] rather than on linguistically interpretable patterns. Why does fine-tuning increase the attention to [CLS], but not to [SEP]? As [SEP] may serve as "no-op" indicator, fine-tuning basically may tell BERT what to ignore. In multilingual BERT, fine-tuning affected both the top and the middle layers of the model.

Studies explored the possibilities of improving the fine-tuning of BERT by taking more layers into account: combining deeper layers with the output layer or a weighted representation of all layers; two-stage fine-tuning with an intermediate supervised training stage; adversarial token perturbations that improve robustness of the model; or mixout regularization, which improves the stability of BERT fine-tuning even for a small number of training examples.

With larger and larger models even fine-tuning becomes expensive, but this cost has been limited by adapter modules, which have been also used for multi-task learning and cross-lingual transfer; by reusing monolingual BERT weights for cross-lingual transfer; or by extracting features from frozen representations.

Initialization can have a dramatic effect, which is not often reported: performance improvements claimed in many NLP modeling papers may be within the range of that variation. Significant variation has been reported for BERT fine-tuned on GLUE: both weight initialization and training data order contribute to the variation. Some authors propose an early-stopping technique to avoid full fine-tuning for the less-promising seeds.

4.3.2.6 *How big should BERT be?*

OVERPARAMETRIZATION Transformer-based models keep increasing in size, e.g. T5 is over 30 times larger than the base BERT. This raises concerns about the computational complexity of self-attention, environmental issues, and reproducibility and access to research resources in academia vs. industry. Current models do not make good use of the parameters: all but a few Transformer heads can be pruned without much loss in performance, most BERT heads in the same layer show similar self-attention patterns, and most layers can be reduced to a single head.

Depending on the task, there may be harmful BERT heads/layers. For machine translation and GLUE tasks, both heads and layers could be advantageously disabled. In a structural probing classifier, 5 out of 8 probing tasks show some layers (typically the final one) to cause a drop in scores. Comparing BERT-base and BERT-large, the larger model performs better many times, but the opposite was observed for subject-verb agreement and sentence subject detection. Why does BERT end up with redundant heads and layers? It is not clear given the complexity of language, and amounts of pre-training. The reason was suggested to be the use of attention dropouts.

COMPRESSION BERT can be efficiently compressed with minimal accuracy loss. In a knowledge distillation framework, a smaller student-network is trained to mimic the behavior of BERT. Variants include mimicking the activation patterns of individual portions of the teacher, and knowledge transfer at different stages (pre-training or fine-tuning). Another method is quantization of weights, which often requires compatible hardware. Other techniques include decomposing BERT's embedding matrix into smaller matrices.

PRUNING AND MODEL ANALYSIS Care has to be taken in linguistic analysis. For example, BERT has heads that seem to encode frame-semantic relations, but disabling them might not hurt downstream task performance, which suggests that this knowledge is not actually used. A study identified the functions of self-attention heads and then checked which of them survive the pruning, finding that syntactic and positional heads are the last ones to go. In the opposite direction is pruning on the basis of importance scores, and interpreting the remaining "good"

subnetwork. It does not seem to be the case that only the heads that potentially encode nontrivial linguistic patterns survive the pruning.

Models and methodology in these studies differ, so the evidence is inconclusive. Head and layer ablation studies have limitations: they inherently assume that certain knowledge is contained in heads/layers despite evidence of more diffuse representations spread across the full network, i.e. the gradual increase in accuracy on difficult semantic parsing tasks, and the absence of heads that do parsing "in general". Ablating individual components may harm the weight-sharing mechanism, and ablations are also problematic if information is duplicated in the network.

4.3.2.7 *Multilingual BERT*

In version 1 of the paper, Rogers, Kovaleva, and Rumshisky (2020) discuss the Multilingual BERT (mBERT), which has been trained on Wikipedia in 104 languages (with a 110K wordpiece vocabulary). Languages with a lot of data were subsampled, and some were supersampled. mBERT is surprisingly good in zero-shot transfer on many tasks, but not in language generation. It has been used to create high-quality cross-lingual word alignments, with caution for open-class parts of speech. Adding more languages does not seem to harm the quality of representations. mBERT transfers knowledge across some scripts, and retrieves parallel sentences, although it has been noted that this task could be solvable by simple lexical matches. The representation space shows some systematicity in between-language mappings. "Translation" is possible by shifting the representations by a sentences offset. However, mBERT does not learn systematic transformations of structures to accommodate a target language with different word order, e.g. SOV instead of SVO, or a different adjective/noun order.

mBERT is simply trained on a multilingual corpus, with no language IDs, but it encodes language identities. Adding the IDs in pre-training was not beneficial. It reflects at least some typological language features, and transfer between structurally similar languages works better. This implies that mBERT could not be considered as interlingua, because its representation space is structured by typological features. Cross-lingual transfer can be achieved by only retraining the input embeddings while keeping monolingual BERT weights, i.e. even monolingual models learn generalizable linguistic abstractions. Compared with English BERT, at least some of the syntactic properties hold for mBERT: MLM is aware of 4 types of agreement in 26 languages, and the main auxiliary of the sentence can be detected in German and Nordic languages.

There have been conflicting results whether shared word-pieces help mBERT. The simplest formalization of this question is whether performance correlates with the amount of shared vocabulary. Proposals for improving mBERT include fine-tuning on multilingual datasets by freezing the bottom layers; improving word alignment in fine-tuning;

translation language modeling as an alternative pre-training objective where words are masked in parallel sentence pairs; and combining 5 pre-training tasks (monolingual and cross-lingual MLM, translation language modeling, cross-lingual word recovery and paraphrase classification). The monolingual BERT has been applied directly in cross-lingual setting, by initializing the encoder part of the neural MT model with monolingual BERT. mBERT does not have to be pre-trained on multiple languages: you can freeze the Transformer weights and retrain only the input embeddings.

4.3.2.8 *Limitations*

It has been aptly stated that "the fact that a linguistic pattern is not observed by our probing classifier does not guarantee that it is not there, and the observation of a pattern does not tell us how it is used". Furthermore, there is a trade-off between the complexity of the probe and the tested hypothesis: a more complex probe might be able to recover more information, but it is less clear whether we are still talking about the orig model. Finally, different probing methods may reveal contradictory information: certain methods might also favor a certain model, e.g. RoBERTa and BERT with two tree extraction methods.

4.3.2.9 *Directions for further research*

BERT was shown to rely on shallow heuristics in natural language inference, reading comprehension, argument reasoning comprehension, and text classification. Such heuristics can even be used to reconstruct a non-publicly-available model, suggesting a shortcut in the data. It has been realized in the past years that development of harder *datasets that require verbal reasoning* should be as valued as modeling work. "Amnesic probing" targets what knowledge actually gets used by identifying features that are important for prediction for a given task.

4.3.3 *The geometry of word senses*

Coenen et al. (2019) discover separate semantic and syntactic subspaces in BERT representations: a fine-grained geometric representation of word senses, and syntactic representations in attention matrices and individual word embeddings. In this section, we summarize the former, i.e. their finding that BERT distinguishes word senses at a very fine level. Much of this information in encoded in a relatively low-dimensional subspace.

The operation of BERT has the following components:

- the input to BERT is based on a sequence of tokens (words or pieces of words),

- the output is a sequence of vectors, one for each input token, a contextualized embedding, and
- the internals consist of two parts. Th initial embedding for each token is created by combining a pre-trained wordpiece embedding with position and segment information; and the initial sequence of embeddings is run through multiple transformer layers producing a new sequence of context embeddings at each step. In each transformer layer is a set of attention matrices, one for each attention head, and each head contains a scalar value for each pair of tokens.

Context embeddings in BERT and related models contain enough information to perform tasks in the NLP pipeline with simple classifiers (linear or small MLP models). Such single global linear transformations have been termed "structural *probes*" (Belinkov et al. 2017b; Conneau et al. 2018; Hewitt and Manning 2019).

4.3.3.1 *Visualization of word senses*

Taking sentences from the introductions to English-language Wikipedia articles, for individual words, they retrieved 1,000 sentences, and visualized the corresponding BERT-base context embeddings using UMAP. With the example of *die*, they find crisp, well-separated clusters: the German article, 'stop living', and the game tool. Within 'stop living', there is a kind of quantitative scale, related to the number of people dying. They ask the questions whether it is possible to find quantitative corroboration that word senses are well-represented; and the seeming contradiction: whether the positions represent syntax or semantics.

4.3.3.2 *Measurement of word sense disambiguation capability*

Coenen et al. train a simple classifier on BERT's internal representations for WSD following the procedure described by Peters, Neumann, Iyyer, et al. (2018), i.e. a nearest-neighbor classifier, considering centroids of a given word sense's BERT-base embeddings in the training data. They achieve a higher F1 score than the previous state of the art, with accuracy monotonically increasing through the layers. An even higher score was obtained using the technique in next paragraph.

4.3.3.3 *WSD in a 128-dimensional space*

Coenen et al. hypothesize a linear transformation under which distances between embeddings would better reflect their semantic relationships. They trained a probe following Hewitt and Manning (2019)'s methodology, i.e. a matrix $B \in R^{k \times m}$, testing different values for m. The loss is, roughly, defined as the difference between the average cosine similarity between embeddings of words with different senses, and that between embeddings of the same sense. In evaluation on WSD, untransformed

BERT embeddings achieve a state-of-the-art accuracy rate of 71.1%. Trained probes achieve slightly improved accuracy down to $m = 128$. Regarding layers, there is only a modest improvement in accuracy for final-layer embeddings. The method more dramatically improves the performance of embeddings at earlier layers: there is much semantic information in the geometry of earlier layers. The finding offers a resolution to the seeming contradiction mentioned above: syntax and semantics reside in separate complementary subspaces.

4.3.4 *Self attention entropy and ambiguous nouns*

NMT has achieved new state-of-the-art performance in translating ambiguous words. Tang, Sennrich, and Nivre (2019) is interested in which component dominates disambiguation. They consider hidden states, and investigate the distributions of self-attention, training a classifier to predict whether a translation is correct given the representation of an ambiguous noun. They find that encoder hidden states outperform (static) word embeddings significantly, which indicates that encoders adequately encode relevant information for disambiguation. In contrast to encoders, the effect of decoder differs by models. Most interestingly, attention weights and attention entropy show that self-attention can detect ambiguous nouns and distribute more attention to the context.

Tang, Sennrich, and Nivre train a classifier which is fed a representation of ambiguous nouns and a word sense (represented as the embedding of a translation candidate). The classifier has to predict whether the two representations match.

They compare word embeddings and encoder hidden states at different layers both from RNNS2S (Luong+ 2015) and Transformer (Vaswa17). Tang, Sennrich, and Nivre find the following.

- Encoders encode lots of relevant information for WSD into hidden states, even in the first layer. The higher the encoder layer, the more relevant information is encoded.
- Forward RNNs are better than backward RNNs in modeling ambiguous nouns.
- Decoders hidden states have different effects on WSD in Transformer and RNNS2S.
- Self-attention focuses on the ambiguous nouns themselves in the first layer and, keeps extracting relevant information from the context in higher layers.
- Self-attention can recognize the ambiguous nouns and distribute more attention to the context words compared to dealing with nouns in general.

4.3.5 *Psycholinguistic diagnostics*

Ettinger (2020) introduces a suite of diagnostics drawn from psycholinguistic experiments, that allow us to ask targeted questions about the information used by LMs. The results are that BERT can generally distinguish good sentence completions from bad ones involving shared category or role reversal, albeit with less sensitivity than humans; it robustly retrieves noun hypernyms; but struggles with challenging inferences and role-based event prediction with a clear insensitivity to the contextual impacts of negation. She is conservative in the conclusion because these sets are small, and different formulations may yield different performance.

Her diagnostics target a range of linguistic capacities, drawn from psycholinguistics (but she does not test whether LMs are psycholinguistically plausible). The psycholinguistic origin of the test has advantages: it is carefully controlled to ask targeted questions about linguistic capabilities, it asks the questions by examining word predictions in context, which is natural in the LM paradgm, and it allows us to study LMs without any need for task-specific fine-tuning. These diagnostics are chosen specifically to reveal insensitivities in predictive models, as evidenced by patterns that they elicit in human brain responses (N400). They go beyond the syntactic focus seen in existing LM diagnostics, and target commonsense/pragmatic inference, semantic roles and event knowledge, category membership, and negation.

Each of Ettinger's diagnostics is set up to support tests of word prediction accuracy and sensitivity to distinctions between good and bad context completions. Ettinger focus on the BERT model, but the diagnostics are applicable for testing of any LM. Her main contributions are a new set of targeted diagnostics for assessing linguistic capacities that shed light on strengths and weaknesses of the popular BERT model.

4.3.5.1 *Related Work*

The related work section includes work on fine-grained classification tasks to probe information in sentence embeddings, token-level and other sub-sentence level information in contextual embeddings, specific linguistic phenomena such as function words, the overall level of "understanding" (semantic similarity and entailment), and curated versions of these tasks to test for specific linguistic capabilities. The analysis of linguistic capacities of LMs has been dominated by syntactic testing.

Going into details, the *internal dynamics* underlying LMs' capturing of syntactic information has been examined in different components of the LM and at different timesteps within the sentence, in individual units, and regarding semantic phenomena like negative polarity items (analysis still firmly rooted in the notion of detecting structural dependencies). *Word prediction* accuracy has been applied as a test of LMs' language understanding with the LAMBDA dataset, which tests a mod-

els' ability to predict the final word of a passage, in cases where the final sentence alone is insufficient to do so. LAMBDA is not controlled to isolate and test the use of specific types of information.

The linguistic characteristics of the *BERT model itself* has also been examined. Regarding the dynamics of BERT's self-attention mechanism, probing attention heads for syntactic sensitivity found that individual heads specialize strongly for syntactic and coreference relations. Syntactic awareness in BERT has been also examined by syntactic probing at different layers, and examination of syntactic sensitivity in the self-attention mechanism. A variety of linguistic tasks have been tested at different layers. BERT has been found to exhibit very strong performance on several of the targeted syntactic evaluations.

4.3.5.2 *Leveraging psycholinguistic studies*

The fourth section in Ettinger provides background on human language processing, and explains how she uses this information to choose the tests. Psychologists test human responses to words in context, in order to better understand the information that our brains use to generate predictions. Two types of predictive human responses are relevant to us here.

In the *cloze* test, humans are given an incomplete sentence and tasked with filling their expected word in the blank. This is the ideal human prediction in context, not under any time pressure, so participants have the opportunity to use all available information from the context.

The brain response *N400* can be detected by measuring electrical activity at the scalp by EEG to gauge how expected a word is in a context. The electrical signal appears to be sensitive to the fit of a word in context. It correlates with cloze in many cases, it can be predicted by LM probabilities, and, importantly, expectations reflected in the N400 sometimes deviate from the more fully-formed expectations reflected in the untimed cloze response.

Ettinger's draw diagnostic tests from human studies that have revealed divergences between cloze and N400 profiles, i.e. when N400 response suggests a level of insensitivity to certain information in computing expectations, causing a deviation from the fully-informed cloze predictions. These present particularly challenging prediction tasks, tripping up models that fail to use the full set of available information.

4.3.5.3 *Datasets*

Each of Ettinger's diagnostics support three types of testing: word prediction accuracy, sensitivity testing, and qualitative prediction analysis. These diagnostics are constructed to constrain the information relevant for making word predictions. In *word prediction* evaluation accuracy, Ettinger use the most expected items from human cloze probabilities as the gold completions. In what she calls *sensitivity testing*, Ettinger

compares model probabilities for good versus bad completions— specifically, those on which the N400 showed reduced human sensitivity. The question is whether LMs will show similar insensitivities. The *qualitative analysis of models' top predictions* is also informative, because these items are constructed in a controlled manner.

In all tests, the target word to be predicted falls in the final position, which fits the computational models, either left-to-right or bidirectional ones, only token probabilities in context are concerned, and the method is equally applicable to the masked LM setting of BERT as to a standard LM. Ettinger filters out items for which the expected word is not in BERT's single-word vocabulary.

The observations, which we already summarized at the beginning, are based on the following data-sets:

CPRAG-102 (COMMONSENSE AND PRAGMATIC INFERENCE) tests sensitivity to differences within semantic category. In the example *He complained that after she kissed him, he couldn't get the red color off his face. He finally just asked her to stop wearing that lipstick/mascara.*, commonsense knowledge informs us that red color left by kisses suggests lipstick, and pragmatic reasoning allows us to infer that the thing to stop wearing is related to the complaint.

As in LAMBDA, the final sentence is not supporting prediction on its own, but unlike LAMBDA, these items have consistent structure. None of these items contain the target word in context, to test commonsense inference rather than coreference. The average Human cloze probabilities for expected completions is .74. A psycholinguistic study found that inappropriate completions (e.g., *mascara, bracelet*) had cloze probabilities of virtually zero, but N400 showed some expectation for completions that shared a semantic category with the expected completion (e.g., *mascara*, by relation to *lipstick*).

ROLE-88 tests event knowledge and the sensitivity to semantic role reversals, e.g. *The restaurant owner forgot which customer/waitress the waitress/customer had served.* It requires event knowledge about typical interactions between types of entities in the given roles. The authors found that although each completion (e.g., *served*) is good for only one of the noun orders and not the reverse, the N400 shows a similar level of expectation for the target completions regardless of noun order. The sensitivity test targets this distinction. Cloze probabilities show strong sensitivity to the role reversal, with average cloze difference of .233 between good and bad contexts.

NEG-136 tests negation along with knowledge of category membership, e.g. *A robin is (not) a bird/tree.* N400 shows more expectation for true completions in affirmative sentences, but it fails to adjust to negation: There is more expectation for false continuations.

A separate psycholinguistic experiment chose affirmative and negative sentences to be more "natural", e.g. *Most smokers find that quitting is (not) very difficult/easy.*, and contrasts these with affirmative and negative sentences chosen to be less natural *Vitamins and proteins are (not) very good/bad.*

4.3.6 *Layers and lexical content*

Wang and Kuo (2020) generate sentence representation from BERT-based word models exploiting that different layers of BERT capture different linguistic properties. The task of sentence embedding, i.e. transforming a sentence to a vector, is not trivial. A common approach with BERT-based models is to average the representations obtained from the last layer or using the [CLS] token. The authors show that both are sub-optimal. They fuse information across layers to find better sentence representation: Wang and Kuo dissect BERT-based word models through a geometric analysis of the space in an unsupervised fashion.

Different layers of BERT learn different abstraction levels: intermediate layers encode the most transferable features, and higher layers are more expressive in high-level semantic information. Information fusion across layers has great potential. Wang and Kuo experiment on patterns of the isolated word representation across layers, and find that the evolution of isolated word representation patterns across layers highly correlate with word content: words of richer information have higher variation in their representations. This finding helps them define "salient" word representations and informative words for sentence embeddings.

Wang and Kuo compare SBERT-WK with the following 10 (parameterized and non-parameterized) methods: average of GloVe word embeddings; average of FastText word embedding; average the last layer token representations of BERT; [CLS] embedding from BERT, originally used for next sentence prediction; SIF model (Arora, Liang, and Ma 2017), which is a non-parameterized model, a strong baseline in textual similarity tasks; p-mean model that incorporates multiple word embedding models; Skip-Thought; InferSent with both GloVe and FastText versions; Universal Sentence Encoder, which is a strong parameterized sentence embedding using multiple objectives and transformer architecture; and Sentence-BERT, which is a SOTA sentence embedding model with a Siamese network over BERT. SBERT-WK improves the performance on textual similarity tasks by a significant margin. Regarding supervised downstream tasks, SBERT-WK obtains the best result in 5 of the 9 considered tasks, and also in average. The merit of the model is in part due to its efficiency.

Part II

MAIN CONTRIBUTIONS

The second part of this thesis describes the main contributions related to (possibly interlingual) *lexical relations* – relations that hold between the meanings of words independent of context. Chapter 5 starts with word analogies, translation, antonymy (opposite meaning), causality, and hypernymy (what basic category a word belongs to, e.g. *dogs* are *animals*)[5]. Chapter 6 and Chapter 7 investigates the semantic (or thematic) *roles of verb arguments* in symbolic and a distributional perspective, respectively. Chapter 8 concludes the thesis with an evaluation proposal in the distributional study of *word ambiguity*.

5 Sections in this chapter have appeared in proceedings of conferences, and here they appear in that same chronological order.

Nekem szavakról szavak jutnak az eszembe és viszont.
'Words remind me of words and vice versa'

— Péter Esterházy

5 LEXICAL RELATIONS

Contents

In this chapter, we experiment with extracting lexical relations from text corpora in the form of word embeddings. The analized relations include word analogies Section 5.1, translation Section 5.3, antonymy (opposite meaning, Section 5.4), causality Section 5.5, and hypernymy (what basic category a word belongs to, e.g. *dogs* are *animals*, Section 5.6). Sections in this chapter have appeared in proceedings of conferences, and here they appear in that same chronological order.

This line of research is also related to semantic networks. In `4lang`, lexical decomposition is formalized in by 0-edges like `dog` $\xrightarrow{0}$ `animal`, but the most information is apparently included in binary relations like `cow` $\xleftarrow{1}$ `make` $\xrightarrow{2}$ `milk`. The utility of word definitions depends on whether these binary relations capture the right pieces of information. Word embeddings can provide complementary information on whether a putative relations relly exists.

5.1 VECTOR SPACE ANALOGIES

In the last decade, deep neural networks have taken over the state of the art in many areas of artificial intelligence including vision (Krizhevsky and Sutskever 2012), speech processing (Dahl et al. 2011), and language (Peters, Neumann, Iyyer, et al. 2018), reducing the error by a respectable factor. The first wave of a revolution have been word embeddings, word models learned by neural networks, which became very popular since Mikolov, Chen, et al. (2013) and Mikolov, Sutskever, et al. (2013). These more accurate variants of earlier VSMs map "similar" word to similar vectors in space of some hundred dimensions. Word similarity includes syntactic and semantic one, and vector similarity is mostly measured by cosine similarity. Most relevantly to the present chapter, embeddings reflect analogical relations[1] (Mikolov, Yih, and Zweig 2013) like

$$\textbf{woman} - \textbf{man} \approx \textbf{queen} - \textbf{king}$$

5.2 A HUNGARIAN ANALOGICAL BENCHMARK

In Section 4.2, we introduced `word2vec` and GloVe as the two most successful open-source tools that compute distributed language models from gigaword corpora. `word2vec` implements the neural network style architectures skip-gram and cbow, learning parameters using each word as a training sample, while GloVe factorizes the cooccurrence-matrix (or more precisely a matrix of conditional probabilities) as a whole. In this and the following section, which originally appeared as Makrai (2015) and Makrai (2016a), we compare the two systems on two tasks respectively: a Hungarian equivalent of a popular word analogy task and word translation between European languages including medium-resourced ones: Hungarian, Lithuanian and Slovenian.

5.2.1 *Introduction*

The empirical support for both the syntactic properties and the meaning of a word form consists in the probabilities with that the word appears in different contexts. Contexts can be documents as in latent semantic analysis (LSA, Section 4.1.3) or other words appearing within a limited distance (window) from the word in focus. In these approaches, the corpus is represented by a matrix with rows corresponding to words and columns to contexts, with each cell containing the conditional probability of the given word in the given context. The matrix has to undergo some regularization to avoid overfitting. In LSA this is achieved by approximating the matrix as the product of special matrices.

1 This propoerty is called relational similarity by (Levy and Goldberg 2014b).

Neural nets are taking over in many filed of artificial intelligence. In natural language processing applications, training items are the word tokens in a text. Vectors representing word forms on the so called embedding layer have their own meaning: Collobert and Weston 2008 trained a system providing state of the art results in several tasks (part of speech tagging, chunking, named entity recognition, and semantic role labeling) with the same embedding vectors. Mikolov, Yih, and Zweig (2013) trained an embedding with the skip-gram (sgram) architecture that not only encodes similar word with similar vectors but reflects *relational similarities* (similarities of relations between words) as well. The system answers analogical questions. For more details see Section 5.2.2.

The two approaches, one based on cooccurrence matrices and the other on neural learning are represented by the two leading open-source tools for computing distributed language models (or simply vector space language models, VSM) from gigaword corpora, GloVe and `word2vec` respectively. Here we compare them on a task related to statistical machine translation. The goal of the EFNILEX project has been to generate protodictionaries for European languages with fewer speakers. We have collected translational word pairs between English, Hungarian, Slovenian, and Lithuanian

We took the method of Mikolov, Le, and Sutskever (2013) who train VSMs for the source and the target language from monolingual corpora, and collect word translation by learning a mapping between these supervised by a seed dictionary of a few thousand items.

Before collecting word translations, we test the models in an independent and simpler task, the popular analogy task. For this, we created the Hungarian equivalent of a the test question set by Mikolov, Yih, and Zweig (2013) and Mikolov, Chen, et al. (2013).[2]

The only related work evaluating vector models of a language other than English on word analogy tasks we know is Sen and Erdogan (2014) who compare different strategies to deal with the a morphologically rich Turkish language[3] As far as we know, application of GloVe to word translations was a novelty of Makrai (2015).

5.2.2 *Monolingual analogical questions*

Measuring the quality of VSMs in a task-independent way is motivated by the idea of representation sharing. VSMs that capture something of language itself are better that ones tailored for the task. We compare results in the monolingual and the main task in Section 5.3.2.4.

Analogical questions (also called relational similarities (Turney 2006) or linguistic regularities (Mikolov, Yih, and Zweig 2013)) are such a

2 For data and else visit the project page `http://corpus.nytud.hu/efnilex-vect`.
3 I'm grateful to Mehmet Umut Sen for translating the essence of Sen and Erdogan (2014) to English.

English		Hungarian	
plural	singular	plural	singular
decrease	decreases	lesznek	lesz
describe	describes	állnak	áll
eat	eats	tudnak	tud
enhance	enhances	kapnak	kap
estimate	estimates	lehetnek	lehet
find	finds	nincsenek	nincs
generate	generates	kerülnek	kerül

Table 10: Morphological word pairs

measure of merit for vector models. This test has gained popularity in the VSM community in the recent year. Mikolov et al. observe that analogical questions like *good* is to *better* as *rough* is to ... or *man* is to *woman* as *king* is to ... can be answered by basic linear algebra in neural VSMs:

$$\text{good} - \text{better} \approx \text{rough} - \mathbf{x} \tag{1}$$

$$\mathbf{x} \approx \text{rough} - \text{good} + \text{better} \tag{2}$$

So the vector nearest to the right side of (2) is supposed to be *rougher*, which is really the case.

We created a Hungarian equivalent of the analogical questions made publicly available by Mikolov, Yih, and Zweig (2013) and Mikolov, Chen, et al. (2013). More precisely, we follow the main ideas reported in Mikolov, Yih, and Zweig (2013), and target the sizes of the data-set accompanying Mikolov, Chen, et al. (2013).

Analogical pairs are divided to morphological ("grammatical") and semantic ones. The morphological pairs in Mikolov, Yih, and Zweig (2013) were created in the following way:

> [We test] base/comparative/superlative forms of adjectives; singular/plural forms of common nouns; possessive/non-possessive forms of common nouns; and base, past and 3rd person present tense forms of verbs. More precisely, we tagged 267M words of newspaper text with Penn Treebank POS tags (Marcus, Santorini, and Marcinkiewicz 1993). We then selected 100 of the most frequent comparative adjectives (words labeled JJR); 100 of the most frequent plural nouns (NNS); 100 of the most frequent possessive nouns (NN POS); and 100 of the most frequent base form verbs (VB).

	English		Hungarian
	# questions	# pairs	# questions
gram1-adjective-to-adverb	32	992	40
gram2-opposite	812	29	30
gram3-comparative	37	1332	40
gram4-superlative	34	1122	40
gram5-present-participle	33	1056	40
gram6-nationality-adjective	41	1599	41
gram7-past-tense	40	1560	40
gram8-plural-noun	37	1332	40
gram9-plural-verb	30	870	40
capital-common-countries	23	506	20
capital-world	116	4524	166
city-in-state	2467	68	
county-center			19
county-district-center			175
currency	30	866	30
family	23	506	20

Table 11: Sizes of the question sets

English		Hungarian	
Athens	Greece	Budapest	Magyarország
Baghdad	Iraq	Moszkva	Oroszország
Bangkok	Thailand	London	Nagy-Britannia
Beijing	China	Berlin	Németország
Berlin	Germany	Pozsony	Szlovákia
Bern	Switzerland	Helsinki	Finnország
Cairo	Egypt	Bukarest	Románia

Table 12: Semantic word pairs

English				Hungarian			
Athens	Greece	Baghdad	Iraq	Budapest	Magyarország	Moszkva	Oroszország
Athens	Greece	Bangkok	Thailand	Budapest	Magyarország	London	Nagy-Britannia
Athens	Greece	Beijing	China	Budapest	Magyarország	Berlin	Németország
Athens	Greece	Berlin	Germany	Budapest	Magyarország	Pozsony	Szlovákia
Athens	Greece	Bern	Switzerland	Budapest	Magyarország	Helsinki	Finnország
Athens	Greece	Cairo	Egypt	Budapest	Magyarország	Bukarest	Románia

Table 13: Analogical questions

The Hungarian morphological pairs were created in the following way: For each grammatical relationship, we took the most frequent inflected forms from the Hungarian Webcorpus (Halácsy et al. 2004). The suffix in question was restricted to be the last one. See sizes in Table 11. In the case of opposite, we restricted ourselves to forms with the derivational suffix *-tlan* (and its other allomorphs) to make the task morphological rather then semantic. plural-noun includes pronouns as well.

For the semantic task, data were taken from Wikipedia. For the capital-common-countries task, we choose the one-word capitals appearing in the Hungarian Webcorpus most frequently. The English task city-in-state contains USA cities with the states they are located in. The equivalent tasks `county-center` contains counties (*megye*) with their centers (*Bács-Kiskun – Kecskemét*) currency contains the currencies of the most frequent countries in the Webcorpus. The `family` task targets gender distinction. We filtered the pairs where the gender distinction is sustained in Hungarian (but dropping e.g. *he – she*). We put some relational nouns in the possessive case (*bátyja – nővére*). We note that this category contains the royal "family" as well, e.g. the famous *king – queen*, and even *policeman – policewoman*.

Both morphological and semantic questions were created by matching every pair with every other pair resulting in e.g. $\binom{20}{2}$ questions for family.

5.3 WORD TRANSLATION IN EUROPEAN LANGUAGES

For collection of word translations, we take the method of Mikolov, Le, and Sutskever (2013) that starts with creating a VSM for the source and the target language from monolingual corpora in the magnitude of billion(s) of words. VSMs represent words in vector spaces of some hundred dimensions. The key point of the method is learning a linear mapping from the source vector space to the target space supervised by a seed dictionary of 5 000 words. Training word pairs are taken from among the most frequent ones skipping pairs with a source of target word unknown to the language model. The learned mapping is used to find a translation for each word in the source model. The computed translation is the target word with a vector closest to the image of the source word vector by the mapping. The closeness (cosine similarity) between the image of the source vector and the closest target vector measures the goodness of the translation, the similarity of the source and the computed target word. Best results are reported when the dimension of the source model is 2–4 times the dimension of the target model, e.g. 800 → 300.

We generate word translations between the following language pairs: Hungarian-Lithuanian, Hungarian-Slovenian, and Hungarian-English.

The method provides a measure of confidence for each translational pair, namely the distance of the vector computed by mapping the source word vector, and the nearest target word vector. This measure makes a tuning between precision and recall possible (Table 19). With a higher cosine similarity cut-off (column cos >), we get word translations for a smaller vocabulary (vocab) with a higher precision, while lower cosine similarities produce a greater vocabulary with translations of a lower precision. prec@1 is the ratio of words, for which the first candidate translation coincides with that provided in the seed dictionary, prec@5 is the ratio of words with the seed translation in the first 5 candidates. These are strict metrics, as synonyms of the gold translation count as incorrect. gold is the number of words with a gold translation in the corresponding part of the test data.

We follow Mikolov, Yih, and Zweig (2013) in using least squares of the Euclidean distance for training, and, surprisingly, cosine similarity for translation generation, which is the only combination of the two distances that works.

5.3.1 *Data*

5.3.1.1 *Corpora and vectors*

For English, we use vector models downloaded from the home pages of the tools, while for the medium-resourced languages, we train new

language	corpus	# words
Lithuanian	webcorpus (Zséder et al. 2012)	1.4 B
Slovenian	slWaC (Ljubešić and Erjavec 2011)	1.6 B
Hungarian	Webcorpus (Halácsy et al. 2004)	0.7 B
Hungarian	HNC (Oravecz, Váradi, and Sass 2014)	0.8 B

Table 14: Corpora for medium-resourced languages. Word counts are given in billions.

models on the corpora in Table 14, using the tokenization provided by the authors of the corpora[4]

5.3.1.2 *Seed dictionaries*

Mikolov, Le, and Sutskever (2013) use Google translate as a seed dictionary. We have been experimenting with three seed dictionaries: (1) efnilex12, the protodictionaries collected within the EFNILEX project (Héja and Takács 2012), (2) word pairs collected using wikt2dict with and without triangulation (See (Ács, Pajkossy, and Kornai 2013), and, for sizes, Table 15), and (3) dictionaries from the opus collection (Europarl, OpenSubtitles2012 and OpenSubtitles2013) (Tiedemann 2012)[5]. efnilex12 contains directed dictionaries (ranked by the conditional probability of the (cooccurrence of the) target word conditioned on the source word).

	efnilex12	wikt	wikt triang	OSub12	OSub13	Europarl
en-hu	83 K	47 K	+134 K	97 K	19 K	21 K
hu-lt	152 K	6 K	+21 K	11 K	9 K	27 K
hu-sl	235 K	2 K	+26 K	63 K	45 K	29 K

Table 15: Number of translational word pairs in the seed dictionaries

5.3.2 *Results*

Throughout the following two subsections, these abbreviations will be used: d for dimension, w for window radius ($w = 15$ means that (a maximum of) 15 words are considered on both sides of the word in focus), i for number of training iterations over the corpus (epochs), m for minimum word count in the vocabulary cutoff, and n for number of negative samples (in the case of `word2vec`.

4 I would like to thank Vladimír Benko for information on corpora.
5 http://opus.lingfil.uu.se/

		morph		semant		total	
en, Mikolov et al (2013)	$n = 5$	61		58		60	
	$n = 15$	61		61		61	
	HS	52		59		55	
hu	$n = 5$	63.0	3419/5430	38.5	269/699	60.2	3688/6129
	$n = 15$	61.9	3359/5430	39.2	274/699	59.3	3633/6129
	HS	48.9	2653/5430	22.5	157/699	45.8	2810/6129

Table 16: Comparison of results in word translations to those of Mikolov, Le, and Sutskever (2013)

5.3.2.1 *Analogical questions*

For comparing the Hungarian analogical questions to the English ones, we trained sgram models on the concatenation of HNC and the Hungarian Webcorpus with $d = 300, m = 5$ comparing negative sampling to hierarchical softmax (two techniques to avoid computing a the denominator of softmax that is a sum with as many terms as there are words in the embedding) and the effect of subsampling of frequent words, see (Mikolov, Sutskever, et al. 2013) for details. In Table 16, it can be seen that we (below the line) get similar results in the Hungarian equivalent of the original tasks ((Mikolov, Sutskever, et al. 2013), above the line) in the morphological questions, while Hungarian results in the semantic questions are worse, suggesting that the semantic questions are too hard. This problem has to be investigated further.

5.3.2.2 *Protodictionary generation*

In this subsection we report our results in Slovenian/Hungarian/Lithuanian to English protodictionary generation. We take four source embeddings: two Slovenian ones trained on slWaC, one trained on the Hungarian Webcorpus, and one on the Lithuanian webcorpus by Zséder et al. (2012), all in $d = 600$. One of the Slovenian models is a GloVe one, the other models are cbow models with $n = 15$ and $w = 10$. The target model is always glove.840B.300d from the GloVe site, the seed dictionary is OpenSubtitles2012. The source (rs), the target (rt) embedding, or both (rst) was restricted to words accepted by Hunspell. In Table 17 we compare our results (below the line) to those of Mikolov, Le, and Sutskever (2013) (above the line) with slightly different metaparameters. The vocabulary cutoff m of the source embedding is specified for each `word2vec` model we trained.

5.3.2.3 *word2vec, LBL4word2vec and GloVe*

We compared `word2vec`, its modification LBL4word2vec[6], and GloVe with two parameter settings in the two tasks. The two parameter set-

6 https://github.com/qunluo/LBL4word2vec

	prec@1	prec@5
en → sp	33	51
sp → en	35	52
en → cz	27	47
cz → en	23	42
en → vn	10	30
vn → en	24	40
glove-sl → en rs	44.80	63.40
word2vec-sl → en $m = 100$ rs	41.70	60.40
word2vec-hu → en $m = 50$ rst	32.80	54.70
word2vec-lt → en $m = 100$ rt	21.20	36.50

Table 17: Results in protodictionary collection

source word	cos	translations			
öt	0.9101	five	six	eight	three
jó	0.8961	good	really	too	very
de	0.8957	but	though	even	just
bár	0.8955	though	but	even	because
hit	0.8904	faith	belief	salvation	truth
ugyan	0.8880	though	but	even	because
vöröshagymát	0.8878	onion	garlic	onions	tomato

Table 18: Example word translations. cos is the cosine similarity of the image of the source word vector by the learned mapping and the nearest target vector. Words in the target language are listed in the (descending) order of their similarity to the image vector.

cos >	vocab	gold	prec@1	prec@5
0.7	3803	301	68.4%	84.4%
0.6	9967	711	54.7%	74.1%
0.5	12949	958	46.6%	65.6%
0.4	13451	988	45.3%	64.0%

Table 19: Trade-off between precision and recall in Hungarian to English word translation.

	word2vec	GloVe
d	**100**	50
w	5	**15**
i	5	**25**
m	**5**	10

Table 20: Default values of parameters shared by word2vec and GloVe

		morph		sem		total	
small	word2vec sgram	49.0%	2703	20.3%	156	45.5%	2859
	LBL4word2vec sgram	46.6%	2567	19.4%	149	43.2%	2716
	word2vec cbow	49.9%	2751	15.7%	121	45.7%	2872
	glove	41.3%	2277	11.1%	85	37.6%	2362
big	word2vec sgram	57.8%	3186	42.0%	323	55.8%	3509
	LBL4word2vec sgram	55.5%	3058	36.3%	279	53.1%	3337
	glove	58.1%	3206	31.3%	241	54.9%	3447
	word2vec cbow	57.8%	3187	30.7%	236	54.5%	3423

Table 21: Comparison of models trained in different architectures. Rows within each model "size" are sorted by precision in the semantic task, which we consider more relevant to lexicography than morphology. The total number of questions that do not contain out-of-vocabulary words is 5514 in morphological questions and 6283 in semantic ones.

tings were needed because the default (recommended) values of d, w, i and m are different in the two architectures, see Table 20 with the more computation-intensive setting in bold. We trained two models with each architecture on HNC: a small one with the less computation-intensive one of the two default values and a big one with the lesser one (except for using $d = 52$ in small for historical reasons). For the number of negative samples, which is specific for word2vec, we use the default $n = 5$. See results in Table 21. Note that GloVe results could be further improved by taking the average of the two vectors, the "focus" and context vector learned by the model for each word (see Section 4.2.6).

5.3.2.4 *Comparison of results in the two tasks*

In Figure 13 we show the results of some Hungarian VSMs in the analogical and the word translation task plotted against each other. The horizontal axis shows precision in the semantic analogical questions, while the vertical axis shows precision (@5) in protodictionary generation to the Google News model[7] restricted to words accepted by Hunspell and using seed pairs collected with wikt2dict. It can be seen that result in the two tasks are unfortunately uncorrelated.

7 https://code.google.com/p/word2vec/#Pre-trained_word_and_phrase_vectors

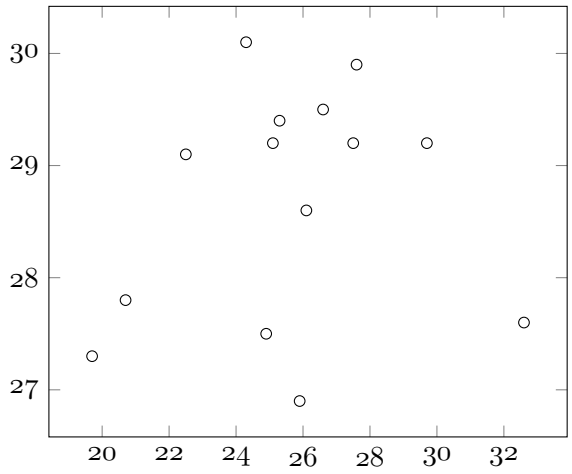

Figure 13: Precision in monolingual (horizontal axis) vs bilingual (vertical axis) task

model	question type	Webcorpus		HNC	
word2vec	morphological	**54.9**	**2924**/5326	51.8	2856/5514
	semantic	8.3	40/482	**16.0**	**123**/769
	total	**51.0**	2964/5808	47.4	**2979**/6283
glove	morphological	47.4	2525/5326	**48.2**	**2658**/5514
	semantic	9.3	45/482	**14.4**	**111**/769
	total	**44.2**	2570/5808	44.1	**2769**/6283

Table 22: Comparison of results on two different corpora. The denominator of each fraction is the number of questions with all three words known to the vector model, while the numerator is the number of correct answers for these questions. Parameters: $d = 152$, $m = 10$, $i = 5$ in both models. For word2vec, $w = 5$ and $n = 5$ while for glove, $w = 3$. The different window sizes mean that these results are not suitable for comparing the models just the corpora.

5.3.3 *Parameter analysis*

5.3.3.1 *Corpus*

QUALITY In Table 22, we compare on analogical questions models trained on the Hungarian National Corpus (September 12 snapshot) (Oravecz, Váradi, and Sass 2014) that is a curated corpus of Hungarian, and on the Hungarian Webcorpus (Halácsy et al. 2004) that is a similarly sized webcorpus. The numbers suggest that a curated corpus is more suitable for the analogical task.

SIZE Table 23 shows how the performance depends on the size of the corpus. It is clear that a much larger corpus is needed to answer semantic questions.

	morph		sem		total	
1M	1.8	58/3256	0.0	0/84	1.7	58/3340
2M	6.1	191/3130	0.0	0/60	6.0	191/3190
10M	24.9	986/3954	7.4	8/108	24.5	994/4062
100M	55.1	2530/4594	31.4	37/118	54.5	2567/4712
754M	63.2	3486/5514	49.8	383/769	61.6	3869/6283

Table 23: The effect of corpus size.

	morph		semant		total	
cbow $hs = 0, n = 5$	59.4%	**3276**/5514	24.1%	185/769	55.1%	3461/6283
cbow $hs = 1, n = 0$	49.0%	2702/5514	13.9%	107/769	44.7%	2809/6283
cbow $hs = 1, n = 5$	49.5%	2730/5514	14.3%	110/769	45.2%	2840/6283
sgram $hs = 0, n = 5$	59.1%	3261/5514	33.6%	**258**/769	56.0%	3519/6283
sgram $hs = 1, n = 0$	49.8%	2744/5514	23.1%	178/769	46.5%	2922/6283
sgram $hs = 1, n = 5$	50.4%	2781/5514	23.1%	178/769	47.1%	2959/6283

Table 24: Hierarchical softmax (HS) and negative sampling.

5.3.3.2 *word2vec*

HIERARCHICAL SOFTMAX AND NEGATIVE SAMPLES We also tried whether hierarchical softmax (HS) and negative sampling can be combined to get better result with either of the techniques. A negative answer can be seen in Table 24 (HNC, $d = 100, w = 5, i = 5, m = 5$).

5.3.3.3 *Protodictionaries: Seed dictionary*

We compare result obtained in the protodictionary generation task with different English-Hungarian seed dictionaries in Table 25. The source language model is always glove.840B.300d[8], the target model is also a GloVe model trained on HNC ($d = 300, m = 1, w = 15, i = 25$). For details of the seed dictionaries see Section 5.3.1.2.

5.4 ANTONYMS IN AN EMBEDDING FROM A DEFINITION GRAPH

In this section, which originally appeared as Makrai, Nemeskey, and Kornai (2013)[9], we test which putative semantic features like GENDER are captured by VSMs. We assume that the difference between two vectors, for antonyms, distills the actual property which is the opposite

8 http://nlp.stanford.edu/projects/glove/

9 Makrai classified the antonymic relation pairs, and prepared the test. Nemeskey finished the experiments. The applicativity idea, which gave the title of the paper, and is regarded here as possible future work, is due to Kornai.

seed dictionary	prec@1	prec@5
Europarl	17.70%	34.10%
wikt triang	13.10%	25.30%
wikt	12.50%	25.40%
OpenSubtitles2012	10.30%	23.40%
efnilex12 en→hu	10.10%	23.80%

Table 25: Accuracy of protodictionary generation with different seed dictionaries

GOOD		VERTICAL	
safe	out	raise	level
peace	war	tall	short
pleasure	pain	rise	fall
ripe	green	north	south
defend	attack	shallow	deep
conserve	waste	ascending	descending
affirmative	negative	superficial	profound
⋮	⋮	⋮	⋮

Table 26: Word pairs associated to features GOOD and VERTICAL

in each member of a pair of antonyms. So, for example, for a set of male and female words, such as ⟨king, queen⟩, ⟨actor, actress⟩, etc., the difference between words in each pair should represent the idea of gender. To test the hypothesis, we associated antonymic word pairs from the WordNet (Miller 1995) to 26 classes e.g. END/BEGINNING, GOOD/BAD, . . . , see Table 26 and Table 28 for examples.

The intuition to be tested is that the first member of a pair relates to the second one in the same way among all pairs associated to the same feature. For k pairs $\vec{x}_i, \vec{y}_i$ we are looking for a common vector $\vec{a}$ such that

$$\vec{x_i} - \vec{y_i} = \vec{a} \tag{3}$$

Given the noise in the embedding, it would be naive in the extreme to assume that (3) can be a strict identity. Rather, our interest is with the best $\vec{a}$ which minimizes the error

$$Err = \sum_i ||\vec{x_i} - \vec{y_i} - \vec{a}||^2 \tag{4}$$

As is well known, E will be minimal when $\vec{a}$ is chosen as the arithmetic mean of the vectors $\vec{x_i} - \vec{y_i}$. The question is simply the following: is the

minimal E_m any better than what we could expect from a bunch of random $\vec{x_i}$ and $\vec{y_i}$?

We selected 26 potentially antonymic datasets from WordNet such as the 'gender' set discussed above. For example, the 'hard' set contains the pairs *hardened/soft, hardball/softball, hardware/software, still/sparkling, hard/soft, solid/gaseous, tough/tender, liquid/gaseous, hardness/softness, hard_drug/soft_drug, hard_water/soft_water* and the 'distance' set contains the pairs *express/local, distant/close, repulsive/attractive, open/close, far/near, distribution/concentration, distributed/concentrated, expanded/contracted, ultimate/proximate, distal/proximal.* Since the sets are of different sizes, we took 100 random pairings of the words appearing on either sides of the pairs to estimate the error distribution, computing the minima of

$$Err_{rand} = \sum_i ||\vec{x_i}' - \vec{y_i}' - \vec{a}||^2 \tag{5}$$

For each distribution, we computed the mean and the variance of Err_{rand}, and checked whether the error of the correct pairing, Err is at least 2 or 3 σs away from the mean.

Table 27 summarizes our results for three embeddings: the original and the scaled HLBL (Mnih and G. E. Hinton 2009) and SENNA (Collobert et al. 2011). The first two columns give the number of pairs considered for a feature and the name of the feature. For each of the three embeddings, we report the error Err of the unpermuted arrangement, the mean m and variance σ of the errors obtained under random permutations, and the ratio

$$r = \frac{|m - Err|}{\sigma}.$$

Horizontal lines divide the features to three groups: for the upper group, $r \geqslant 3$ for at least two of the three embeddings, and for the middle group $r \geqslant 2$ for at least two.

For the features above the first line we conclude that the antonymic relations are well captured by the embeddings, and for the features below the second line we assume, conservatively, that they are not. (In fact, looking at the first column of Table 27 suggests that the lack of significance at the bottom rows may be due primarily to the fact that WordNet has more antonym pairs for the features that performed well on this test than for those features that performed badly, but we did not want to start creating antonym pairs manually.) For example, the putative sets in Table 28 does not meet the criterion and gets rejected.

5.4.1 *Embedding from a definition graph*

The `4lang` embedding is created in a manner that is notably different from the others. Our input is a graph whose nodes are concepts, with

# pairs	feature name	HLBL original *Err*	*m*	σ	r	HLBL scaled *Err*	*m*	σ	r	SENNA *Err*	*m*	σ	r
156	good	1.92	2.29	0.032	11.6	4.15	4.94	0.0635	12.5	50.2	81.1	1.35	22.9
42	vertical	1.77	2.62	0.0617	13.8	3.82	5.63	0.168	10.8	37.3	81.2	2.78	15.8
49	in	1.94	2.62	0.0805	8.56	4.17	5.64	0.191	7.68	40.6	82.9	2.46	17.2
32	many	1.56	2.46	0.0809	11.2	3.36	5.3	0.176	11	43.8	76.9	3.01	11
65	active	1.87	2.27	0.0613	6.55	4.02	4.9	0.125	6.99	50.2	84.4	2.43	14.1
48	same	2.23	2.62	0.0684	5.63	4.82	5.64	0.14	5.84	49.1	80.8	2.85	11.1
28	end	1.68	2.49	0.124	6.52	3.62	5.34	0.321	5.36	34.7	76.7	4.53	9.25
32	sophis	2.34	2.76	0.105	4.01	5.05	5.93	0.187	4.72	43.4	78.3	2.9	12
36	time	1.97	2.41	0.0929	4.66	4.26	5.2	0.179	5.26	51.4	82.9	3.06	10.3
20	progress	1.34	1.71	0.0852	4.28	2.9	3.72	0.152	5.39	47.1	78.4	4.67	6.7
34	yes	2.3	2.7	0.0998	4.03	4.96	5.82	0.24	3.6	59.4	86.8	3.36	8.17
23	whole	1.96	2.19	0.0718	3.2	4.23	4.71	0.179	2.66	52.8	80.3	3.18	8.65
18	mental	1.86	2.14	0.0783	3.54	4.02	4.6	0.155	3.76	51.9	73.9	3.52	6.26
14	gender	1.27	1.68	0.126	3.2	2.74	3.66	0.261	3.5	19.8	57.4	5.88	6.38
12	color	1.2	1.59	0.104	3.7	2.59	3.47	0.236	3.69	46.1	70	5.91	4.04
17	strong	1.41	1.69	0.0948	2.92	3.05	3.63	0.235	2.48	49.5	74.9	3.34	7.59
16	know	1.79	2.07	0.0983	2.88	3.86	4.52	0.224	2.94	47.6	74.2	4.29	6.21
12	front	1.48	1.95	0.17	2.74	3.19	4.21	0.401	2.54	37.1	63.7	5.09	5.23
22	size	2.13	2.69	0.266	2.11	4.6	5.86	0.62	2.04	45.9	73.2	4.39	6.21
10	distance	1.6	1.76	0.0748	2.06	3.45	3.77	0.172	1.85	47.2	73.3	4.67	5.58
10	real	1.45	1.61	0.092	1.78	3.11	3.51	0.182	2.19	44.2	64.2	5.52	3.63
14	primary	2.22	2.43	0.154	1.36	4.78	5.26	0.357	1.35	59.4	80.9	4.3	5
8	single	1.57	1.82	0.19	1.32	3.38	3.83	0.32	1.4	40.3	70.7	6.48	4.69
8	sound	1.65	1.8	0.109	1.36	3.57	3.88	0.228	1.37	46.2	62.7	6.17	2.67
7	hard	1.46	1.58	0.129	0.931	3.15	3.41	0.306	0.861	42.5	60.4	8.21	2.18
10	angular	2.34	2.45	0.203	0.501	5.05	5.22	0.395	0.432	46.3	60	6.18	2.2

Table 27: Error of approximating real antonymic pairs (*Err*), mean and standard deviation (m, σ) of error with 100 random pairings, and the ratio $r = \frac{|Err-m|}{\sigma}$ for different features and embeddings

PRIMARY		ANGULAR	
leading	following	square	round
preparation	resolution	sharp	flat
precede	follow	curved	straight
intermediate	terminal	curly	straight
antecedent	subsequent	angular	rounded
precede	succeed	sharpen	soften
question	answer	angularity	roundness
⋮	⋮	⋮	⋮

Table 28: Features that fail the test

edges running from A to B iff B is used in the definition of A. The base vectors are obtained by the spectral clustering method pioneered by Ng, Jordan, and Weiss (2001): the incidence matrix of the conceptual network is replaced by an affinity matrix whose ij-th element is formed by computing the cosine distance of the ith and jth row of the original matrix, and the first few (in our case, 100) eigenvectors are used as a basis.

Since the concept graph includes the entire Longman Defining Vocabulary (LDV), each LDV element w_i corresponds to a base vector b_i. For the vocabulary of the whole dictionary, we simply take the Longman definition of any word w, strip out the stopwords (we use a small list of 19 elements taken from the top of the frequency distribution), and form $V(w)$ as the sum of the b_i for the w_is that appeared in the definition of w (with multiplicity).

We performed the same computations based on this embedding as in the previous section: the results are presented in Table 29. Judgment columns under the three embeddings in the previous section and `4lang` are highly correlated, see table 30.

Unsurprisingly, the strongest correlation is between the original and the scaled HLBL results. Both the original and the scaled HLBL correlate notably better with `4lang` than with SENNA, making the latter the odd one out.

So far we have seen that a dictionary-based embedding, when used for a purely semantic task, the analysis of antonyms, does about as well as the more standard embeddings based on cooccurrence data. Clearly, a VSM could be obtained by the same procedure from any machine-readable dictionary. Using LDOCE is computationally advantageous in that the core vocabulary is guaranteed to be very small, but finding the eigenvectors for an 80k by 80k sparse matrix would also be within CPU reach.

5.4.2 *Future work: Applicativity*

The main advantage of starting with a conceptual graph lies elsewhere, in the possibility of investigating the function application issue we started out with.

The `4lang` conceptual representation relies on a small number of basic elements, most of which correspond to what are called unary predicates in logic. Kornai (2012) argued that meaning of linguistic expressions can be formalized using predicates with at most two arguments (there are no ditransitive or higher arity predicates on the semantic side). The x and y slots of binary elements such as x *has* y or x *kill* y, Kornai and Makrai (2013) receive distinct labels called NOM and ACC in case grammar (Fillmore 1977); 1 and 2 in relational grammar (Perlmutter 1983); or AGENT and PATIENT in linking theory (Ostler 1979). The label names themselves are irrelevant, what matters

# pairs	feature name	4lang *Err*	4lang *m*	4lang *σ*	4lang *r*
49	in	0.0553	0.0957	0.00551	7.33
156	good	0.0589	0.0730	0.00218	6.45
42	vertical	0.0672	0.1350	0.01360	4.98
34	yes	0.0344	0.0726	0.00786	4.86
23	whole	0.0996	0.2000	0.02120	4.74
28	end	0.0975	0.2430	0.03410	4.27
32	many	0.0516	0.0807	0.00681	4.26
14	gender	0.0820	0.2830	0.05330	3.76
36	time	0.0842	0.1210	0.00992	3.74
65	active	0.0790	0.0993	0.00553	3.68
20	progress	0.0676	0.0977	0.00847	3.56
18	mental	0.0486	0.0601	0.00329	3.51
48	same	0.0768	0.0976	0.00682	3.05
22	size	0.0299	0.0452	0.00514	2.98
16	know	0.0598	0.0794	0.00706	2.77
32	sophis	0.0665	0.0879	0.00858	2.50
12	front	0.0551	0.0756	0.01020	2.01
10	real	0.0638	0.0920	0.01420	1.98
8	single	0.0450	0.0833	0.01970	1.95
7	hard	0.0312	0.0521	0.01960	1.06
10	angular	0.0323	0.0363	0.00402	0.999
12	color	0.0564	0.0681	0.01940	0.600
8	sound	0.0565	0.0656	0.01830	0.495
17	strong	0.0693	0.0686	0.01111	0.0625
14	primary	0.0890	0.0895	0.00928	0.0529
10	distance	0.0353	0.0351	0.00456	0.0438

Table 29: The results on 4lang

	HLBL original	HLBL scaled	SENNA	4lang
HLBL original	1	0.925	0.422	0.856
HLBL scaled	0.925	1	0.390	0.772
SENNA	0.422	0.390	1	0.361
4lang	0.856	0.772	0.361	1

Table 30: Correlations between judgments based on different embeddings

is that these elements are not part of the lexicon the same way as the words are, but rather constitute transformations of the vector space.

Here we will use the binary predicate *x has y* to reformulate the puzzle we started out with, analyzing *queen of England, king of Italy* etc. in a compositional (additive) manner, but escaping the commutativity problem. For the sake of concreteness we use the traditional assumption that it is the king who possesses the realm and not the other way around, but what follows would apply just as well if the roles were reversed. What we are interested in is the asymmetry of expressions like *Albert has England* or *Elena has Italy*, in contrast to largely symmetric predicates. *Albert marries Victoria* will be true if and only if *Victoria marries Albert* is true, but from *James has a martini* it does not follow that *?A martini has James*.

While the fundamental approach of VSM is quite correct in assuming that nouns (unaries) and verbs (binaries) can be mapped on the same space, we need two transformations T_1 and T_2 to regulate the linking of arguments. A form like *James kills* has *James* as agent, so we compute $V(\text{James})+T_1V(\text{kill})$, while *kills James* is obtained as $V(\text{James})+T_2V(\text{kill})$. The same two transforms can distinguish agent and patient relatives as in *the man that killed James* versus *the man that James killed*.

Such forms are compositional, and in languages that have overt case markers, even 'surface compositional' (Hausser 1984). All input and outputs are treated as vectors in the same space where the atomic lexical entries get mapped, but the applicative paradox we started out with goes away. As long as the transforms T_1 and T_2 take different values on *kill, has,* or any other binary, the meanings are kept separate. The interested reader may consult Kornai (2022), who represent irreducible binary elements (e.g. HAS, the comparative ER, CAUSE, the locative AT, etc.) with matrices, and the rest (including transitive verbs) are represented by vectors.

5.5 CAUSALITY IN VECTORS SPACE LANGUAGE MODELS

In this section, which originally appeared as Makrai (2014), we take a semantic relation with rich literature in philosophy and application in knowledge representation, causality (see Figure 14). We are interested in the geometric function mapping a vectors representing some cause (e.g. *hurt*) to the vector representing its effect (*ache*).

First we describe resources, methods, and results. As the results are preliminary, we outline directions of further research as well.

We took causal word pairs from a natural language processing resource containing lexical information of various kinds, WordNet (Miller 1995). The pairs are exemplified in Table 31. We took several VSMs: SENNA (Collobert et al. 2011), (Turian, Ratinov, and Bengio 2010; Huang et al. 2012), HLBL (Mnih and G. E. Hinton 2009), the En-

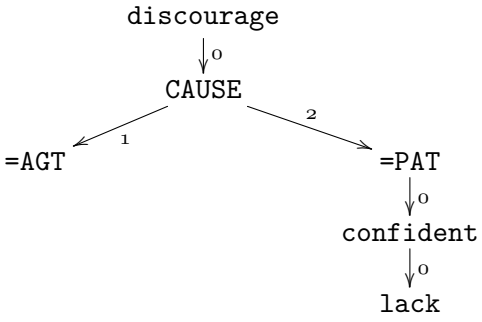

Figure 14: The definition of *discourage* in the `4lang` concept lexicon exemplifies the use of 'cause' in associative network representations of linguistics knowledge. The graph expresses that *discourage* means, that the agent (`=AGT`) causes the participant that is called patient in linguistics (`=PAT`) to lack confidence.

glish Polyglot (Al-Rfou', Perozzi, and Skiena 2013), and 24, variants of the model created from `4lang`. Casual pairs were projected to a 2-dimensional plane by principal component analysis, a machine learning technique often used for visualizing high-dimensional data. The visualization suggested that that there is a center in the vector space representing the words, that approximately fits the lines containing each causal pair, see Figure 15.

For testing the centrality property in the original, unreduced space, we took random word pairs of the same number as we have causal pairs. The point closest to all the lines fitting each pair was computed for both the real and the random sample of word pairs using a formula by Han and Bancroft (2010). Distances of the lines to the corresponding center was also computed. Centrality implies that the expected value of the distances is lower in the real case than in the random case. An unpaired t-test showed that this condition holds in the case of SENNA ($p < 0.001$).

Some of the models created from `4lang` also show significant ($p < 0.05$) difference, but this statistical result has to be taken with caution, because of the phenomenon known as *multiple testing* (Domingos 2012).

> Standard statistical tests assume that only one hypothesis is being tested, but modern learners can easily test millions before they are done. As a result what looks significant may in fact not be. [...] This problem can be combatted by correcting the significance tests to take the number of hypotheses into account [...]

Multiplying the p values by 24 significance is lost, so we should motivate the choice of some specific model among all `4lang` models on some independent grounds to make results significant. This remains a problem for further research.

give	have
show	see
encourage	hope
feed	eat
kill	die
raise	rise
⋮	⋮

Table 31: Word causes and effects in WordNet. WordNet contains semantic relations like is-a (a chair is a furniture), instance-of (Mozart is an instance of 'composer'), antonym (cold and hot), part-of (Monday is a part of 'week') as well.

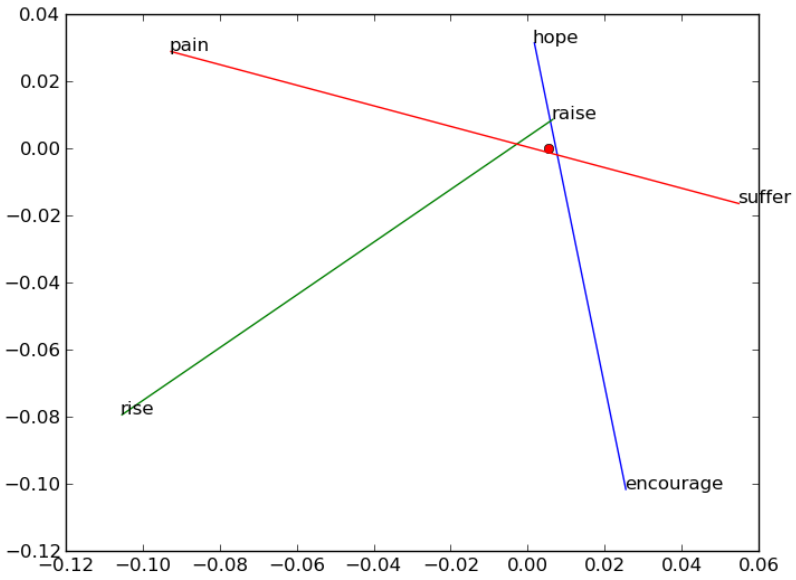

Figure 15: A 2-d visualization of causal pairs in the VSMs suggest that lines connecting causal pairs run close to a common center point.

5.5.1 *Conclusion*

Looking for an insightful interpretation of causality in VSMs, we have found a center point $\mathbf{c}$ in the VSM SENNA with the property that the lines connecting the two members of causal word pairs run close to $\mathbf{c}$. In algebraic terms this means that

$$\mathbf{v}_{\text{effect}} \approx \lambda \mathbf{v}_{\text{cause}} + (1 - \lambda)\mathbf{c}$$

with an appropriate $\lambda \in \mathbb{R}$, reflecting the linguistic intuition that the meaning of the effect is a combination of the meaning of the cause and a causal element.

Further research should be made to discover more sophisticated connections between cause and effect vectors that apply to more models, possibly all models obtained by one or more of the three mentioned methods (co-occurrence matrices, neural nets, and lexicon graphs).[10]

5.6 HYPERNYMY AS INTERACTION OF SPARSE ATTRIBUTES

This section, which originally appeared as Berend, Makrai, and Földiák (2018)[11], describes 300-sparsans' participation in SemEval-2018 Task 9: *Hypernym Discovery*, with a system based on sparse coding and a formal concept hierarchy obtained from word embeddings. Our system took first place in subtasks *(1B) Italian* (*all* and *entities*), *(1C) Spanish entities*, and *(2B) music entities*.

5.6.1 *Introduction*

Natural language phenomena are extremely sparse by their nature, whereas continuous word embeddings employ dense representations of words. Turning these dense representations into a much sparser form can help in focusing on most salient parts of word representations (Faruqui et al. 2015; Berend 2017; Subramanian et al. 2018).

Sparsity-based techniques often involve the coding of a large number of signals over the same dictionary (Rubinstein, Zibulevsky, and Elad 2008). Sparse, over-complete representations have been motivated in various domains as a way to increase separability and interpretability (Olshausen and Field 1997) and stability in the presence of noise.

Non-negativity has also been argued to be advantageous for interpretability (Faruqui et al. 2015; Fyshe et al. 2015; Arora et al. 2016). As Subramanian et al. (2018) illustrates this in the language domain, where sparse features are interpreted as lexical attributes, "to describe the city of Pittsburgh, one might talk about phenomena typical of the

10 I would like to thank Balázs Szalkai for reminding me to the problem of multiple testing.

11 Berend and Makrai worked together and did the same kind of work in the project, but Berend clearly played the role of the first author.

city, like erratic weather and large bridges. It is redundant and inefficient to list negative properties, like the absence of the Statue of Liberty". Berend, Makrai, and Földiák (2018) utilizes non-negative sparse coding for word translation by training sparse word vectors for the two languages such that coding bases correspond to each other.

Here we apply sparse feature pairs to hypernym extraction. The role of an attribute pair $\langle i, j \rangle \in \phi(q) \times \phi(h)$ (where q is the query word, h is the hypernym candidate, and $\phi(w)$ is the index of a non-zero component in the sparse representations of w) is similar to *interaction terms* in regression, see Section 5.6.2 for details.

Sparse representation is related to hypernymy in various natural ways. One of them is through *Formal concept Analysis (FCA)*. The idea of acquiring concept hierarchies from a text corpus with the tools of Formal concept Analysis (FCA) is relatively new (Cimiano, Hotho, and Staab 2005). Our submissions experiment with formal concept analysis tool by Endres, Földiák, and Priss (2010). See the next subsection for a description of formal concept lattices, and how hypernyms can be found in them.

Another natural formulation is related to *hierarchical sparse coding* (Zhao, Rocha, and Yu 2009), where trees describe the order in which variables "enter the model" (i.e. take non-zero values). A node may take a non-zero value only if its ancestors also do: the dimensions that correspond to top level nodes should focus on "general" meaning components that are present in most words. Yogatama et al. (2015) offer an implementation that is efficient for gigaword corpora. Exploiting the correspondence between the variable tree and the hypernym hierarchy offers itself as a natural choice.

The task (Camacho-Collados et al. 2018) evaluated systems on their ability to extract hypernyms for query words in five subtasks (three languages, English, Italian, and Spanish, and two domains, medical and music). Queries have been categorized as concepts or entities. Results were reported for each category separately as well as in combined form, thus resulting in 5×3 combinations. Our system took first place in subtasks *(1B) Italian* (*all* and *entities*), *(1C) Spanish entities*, and *(2B) music entities*. Detailed results for our system appear in Section 5.6.3. Our source code is available online[12].

5.6.1.1 *Formal concept analysis*

Formal concept Analysis (FCA) is the mathematization of *concept* and conceptual hierarchy (Ganter and Wille 2012; Endres, Földiák, and Priss 2010). In FCA terminology, a *context* is a set of *objects* $\mathcal{O}$, a set of *attributes* $\mathcal{A}$, and a binary incidence relation $\mathcal{I} \subseteq \mathcal{O} \times \mathcal{A}$ between members of $\mathcal{O}$ and $\mathcal{A}$. In our application, $\mathcal{I}$ associates a word $w \in \mathcal{O}$ to the indices of its non-zero sparse coding coordinates $i \in \mathcal{A}$. FCA

12 https://github.com/begab/fca_hypernymy

finds formal *concepts*, pairs $\langle O, A\rangle$ of object sets and attribute sets ($O \subseteq \mathcal{O}, A \subseteq \mathcal{A}$) such that A consists of the shared attributes of objects in O (and no more), and O consists of the objects in $\mathcal{O}$ that have all the attributes in A (and no more). (There is a closure-operator related to each FCA context, for which O and A are closed sets iff $\langle O, A\rangle$ is a concept.) O is called the extent and A is the intent of the concept. [13]

There is an order defined in the context: if $\langle A_1, B_1\rangle$ and $\langle A_2, B_2\rangle$ are concepts in C, $\langle A_1, B_1\rangle$ is a *subconcept* of $\langle A_2, B_2\rangle$ if $A_1 \subseteq A_2$ which is equivalent to $B_1 \supseteq B_2$. The concept order forms a complete lattice. The smallest concept whose extent contains a word is said to *introduce* the object. We expect that h will be a hypernym of q iff $n(q) \leqslant n(h)$ where $n(w)$ denotes the node in the concept lattice that introduces w.

The closedness of extents and intents has an important structural consequence. Adding attributes to $\mathcal{A}$ (e.g. responses of additional neurons) will very probably grow the model. However, the original concepts will be embedded as a substructure in the larger lattice, with their ordering relationships preserved.

5.6.2 *Our approach*

Here we describe our system that is based on sparse non-negative word representations and FCA besides more traditional features.

We use the popular skip-gram (SG) approach (Mikolov, Chen, et al. 2013) to train $d = 100$ dimensional dense distributed word representations for each sub-corpus. The word embeddings are trained over the text corpora provided by the shared task organizers with the default training parameters of `word2vec` (w2v), i.e. a window size of 10 and 25 negative samples for each positive context.

We derived *multi-token units* by relying on the `word2phrase` software accompanying the w2v toolkit. An additional source for identifying multi-token units in the training corpora was the list of potential hypernyms released for each subtask by the shared task organizers.

Given the dense embedding matrix $W_x \in \mathbb{R}^{d\times|V_x|}$, for some subcorpus of the shared task $x \in \{1A, 1B, 1C, 2A, 2B\}$, where $|V_x|$ is the size of the vocabulary and d is set to 100. As a subsequent step, we turn W_x into *sparse word vectors* akin to Berend (2017) by solving for

$$\min_{D\in\mathcal{C},\alpha\in\mathbb{R}_{\geqslant 0}} \|D\alpha - W_x\|_F + \lambda\|\alpha\|_1, \tag{6}$$

13 Those who are familiar with closure operators may note that the double application of $'$ is a closure operation both on objects and attributes: with notation $\overline{S} = S''$, for either $S \subseteq \mathcal{O}$ or $S \subseteq \mathcal{A}$, we have $S \subseteq \overline{S}$ and $\overline{\overline{S}} = S$, and the following conditions are equivalent for all $O \subseteq \mathcal{O}$ and $A \subseteq \mathcal{A}$:

- $\langle O, A\rangle$ is a concept
- O is a closed set with respect to $O \mapsto \overline{O}$, and $A = O'$
- A is a closed set with respect to $A \mapsto \overline{A}$, and $O = A'$.

Core feature name	
cosine	$\frac{\mathbf{q}^\intercal \mathbf{h}}{\|\mathbf{q}\|_2 \|\mathbf{h}\|_2}$
difference	$\|\mathbf{q} - \mathbf{h}\|_2$
normRatio	$\frac{\|\mathbf{q}\|_2}{\|\mathbf{h}\|_2}$
qureyBeginsWith	$Q[0] = h$
queryEndsWith	$Q[-1] = h$
hasCommonWord	$Q \cap H \neq \varnothing$
sameFirstWord	$Q[0] = H[0]$
sameLastWord	$Q[-1] = H[-1]$
logFrequencyRatio	$\log_{10} \frac{count(q)}{count(h)}$
isFrequentHypernym[15]	$c \in MF_{50}(q.type)$
sameConcept	$n(h) = n(q)$
parent	$n(q) \prec n(h)$
child	$n(h) \prec n(q)$
overlappingBasis	$\phi(q) \cap \phi(h) \neq \varnothing$
sparseDifference$_{q \setminus h}$	$\|\phi(q) - \phi(h)\|$
sparseDifference$_{h \setminus q}$	$\|\phi(h) - \phi(q)\|$
attributePair$_{ij}$	$\langle i, j \rangle \in \phi(q) \times \phi(h)$

Table 32: The features employed in our classifier. $MF_{50}(q.type)$ refers to the set of top-50 most frequent hypernyms for a given query type.

where $\mathcal{C}$ refers to the convex set of $\mathbb{R}^{d \times k}$ matrices consisting of d-dimensional column vectors with norm at most 1, and α contains the sparse coefficients for the elements of the vocabulary. The only difference compared to Berend (2017) is that here we ensure a non-negativity constraint over the elements of α.

For the elements of the vocabulary we ran the *formal concept analysis* tool of Endres, Földiák, and Priss (2010)[14]. In order to keep the size of the DAG outputted by the FCA algorithm manageable, we only included the query words and those hypernyms in the analysis which occur in the training dataset for the corpora. As we will see in the next subsection, this restriction turns out to be very useful.

Next, we determine a handful of features for a pair of expressions (q, h) consisting of a query q and its potential hypernym h. Table 32 provides an overview of the features employed for a pair (q, h). We denote with $\mathbf{q}$ and $\mathbf{h}$ the 100-dimensional dense vectorial representations of q and h. Additionally, we denote with Q and H the sequence of tokens constituting the query and hypernym phrases. Finally, we

14 www.compsens.uni-tuebingen.de/pub/pages/personals/3/concepts.py

		without attribute pairs						with attribute pairs					
		MAP	MRR	P@1	P@3	P@5	P@15	MAP	MRR	P@1	P@3	P@5	P@15
1A	offic	8.6	18.0	13.0	8.9	8.2	7.9	8.9	19.4	14.9	9.3	8.6	8.1
1A	reprd	9.07	18.7	13.5	9.4	8.8	8.5	9.2	19.9	14.9	9.5	8.7	8.4
1B	offic	9.4	19.9	13.2	9.5	9.3	8.8	12.1	25.1	17.6	12.9	11.7	11.2
1B	reprd	9.2	19.5	12.8	8.9	8.9	8.7	12.8	26.7	18.9	13.6	12.4	11.9
1C	offic	12.5	25.9	16.6	13.6	12.6	11.5	17.9	37.6	27.8	19.7	17.1	16.3
1C	reprd	12.9	26.0	16.2	13.9	13.0	11.9	18.3	38.4	28.5	20.2	17.4	16.6
2A	offic	15.0	32.2	24.8	17.7	15.8	11.6	20.8	40.6	31.6	23.5	21.4	17.1
2A	reprd	15.1	32.4	24.4	18.0	16.2	11.8	21.5	43.7	35.6	25.3	21.8	17.0
2B	offic	19.1	36.7	27.2	23.0	20.1	15.4	29.5	46.4	33.0	31.9	28.9	27.7
2B	reprd	21.5	40.9	29.6	25.6	22.1	18.0	30.4	46.8	33.8	31.8	29.5	28.9

Table 33: Our submissions results: `offic`ial and those that can be `repro`duced with the code in the project repo (with the *isFrequentHypernym* feature being turned off).

refer to the set of basis vectors (in the FCA terminology, attributes) which are assigned non-zero weights in the reconstruction of the vectorial representation of q and h as $\phi(q)$ and $\phi(h)$. It is also considered as a feature (`isFrequentHypernym`) whether a particular candidate hypernym h belongs to the top-50 most frequent hypernyms for the category of q (i.e. concept or entity). Modeling the two categories separately played an important role in the success of our systems.

Three additional features are defined for incorporating the concept lattice output by FCA. With $n(w)$ denoting the concept that introduces w, i.e. the most specific location within the DAG for w, our features indicate whether $n(q)$ (1) coincides with that of h, (2) is the parent (immediate successor) for that of h, or (3) is the child (immediate predictions) for that of h. Parents, and even the inverse relation, proved to be more predictive than the conceptually motivated $q \leqslant h$. In Table 32, $n_1 \prec n_2$ denotes that n_1 is an immediate predecessor of n_2. We will see in post-evaluation ablation experiments, where we refer to the above three features as the *FCA* features, that they were not useful in our submissions.

The `attributePair`$_{ij}$s above, our most important features, are indicator features for every possible interaction term between the sparse coefficients in α. That means that for a pair of words (q, h) we defined $\phi(q) \times \phi(h)$, i.e. candidates get assigned with the Cartesian product derived from the indices of the non-zero coefficients in α. Note that this feature template induces k^2 features, with k being the number of basis vectors introduced in the dictionary matrix D according to Eq. 6.

In order to rank potential hypernym candidates over the test set we trained a *logistic regression* classifier for concepts and entities utilizing the `sklearn` package (Pedregosa et al. 2011)[16] with the regularization parameter defaulting to 1.0.

15 At submission time, this feature did not work properly. MF stands for *most frequent*.
16 scikit-learn.org

For each appropriate (q, h) pair of words for which h is a hypernym of q, we generated a number of *negative samples* (q, h'), such that the training data does not include h' as a valid hypernym for q. For a given query q, belonging to either of the *concept* or *entity* category, we sampled h' from those hypernyms which were included as a valid hypernym in the training data with respect to some $q' \neq q$ query phrase.

When making predictions for the hypernyms of a query, we relied on our query type sensitive logistic regression model to determine the ranking of the hypernym candidates. In our official submission, the ranking was restricted to the phrases which were appeared in the training data as a proper hypernym at least once.

After the appropriate model ranked the hypernym candidates, we selected the top 15 ranked candidates and applied a *post-ranking* heuristic over them, i.e. reordered them according to their background frequency from the training corpus. Our assumption here is that more frequent words tend to refer to more general concepts and more general hypernymy relations potentially tend to be more easily detectable than more special ones.

5.6.3 *Results*

5.6.3.1 *Our submissions*

Our submissions were based on $k = 200$ dimensional sparse vectors computed from unit-normed 100-dimensional dense vectors with $\lambda = .3$. The sum of the two dimensions motivates our group name. For training the regression model with negative samples, 50 false hypernyms were sampled for each query q in the training dataset. One of our submissions involved attribute pairs, the other not. Both submissions used the conceptually motivated but practically harmful FCA-based features.

Table 33 shows submission results. The figures that can be reproduced with the code in the project repo (`reprd`) is slightly different from our official submissions (`offic`) for two reasons: because the implementation of `isFreqHyp` contained a bug, and because of the natural randomness in negative sampling. For reproducibility, we report result without the `isFreqHyp` feature. The randomness introduced by negative sampling is now factored out by random seeding.

5.6.3.2 *Query type sensitive baselining*

Our submission with attribute pairs achieved first place in categories (1B) Italian (all and entities), (1C) Spanish entities, and (2B) music entities. This is in part due to our good choice of a fallback solution in the case of OOV queries: we applied a category-sensitive baseline returning the most frequent train hypernym in the corresponding query type (concept or entity). Table 35 shows how frequently we had to rely

	MAP	MRR	P@1	P@3	P@5	P@10
1A	9.8	22.6	19.8	10.0	9.0	8.6
1A	8.8	21.4	19.8	8.9	7.8	7.5
1B	8.9	21.2	17.1	9.1	8.3	7.9
1B	7.8	19.4	17.1	8.3	6.8	6.5
1C	16.4	33.3	24.6	17.5	16.1	14.9
1C	12.2	29.8	24.6	12.0	11.3	11.0
2A	29.0	35.9	32.6	34.3	34.2	21.7
2A	28.9	35.8	32.6	34.3	34.2	21.4
2B	40.2	58.8	50.6	44.6	40.3	35.5
2B	33.3	51.5	36.2	40.1	35.8	28.4

Table 34: Baseline results, most frequent training hypernyms. We (upper) consider the most frequent hypernym in the given query type (concept or entity). For comparison, we also show the `MFH` baseline provided by the organizers (lower) that is based on the most frequent hypernyms in general.

on this fallback, and Table 34 shows the corresponding pure baseline results.

5.6.3.3 *Post-evaluation analysis*

After the evaluation closed, we conducted ablation experiments the results of which are included in Table 37. In these experiments, we investigated the contribution of the features derived from sparse attribute pairs and FCA. These ablation experiments corroborate the importance of features derived from sparse attribute pairs and reveal that turning off FCA-based features does not hurt performance at all. For this reason – even though our official shared task submission included FCA-related features – we no longer employed them in our post-evaluation experiments.

Table 36 includes the detailed behavior of our model on subtask 1A with respect three distinct factors, that is

1. the number of basis vectors employed during sparse coding ($k \in \{200, 300, 1000\}$),
2. the number of negative training samples per positive sample ($ns \in \{50, all\}$),
3. candidate filtering being turned on/off.

In our original submission we generated 50 negative samples (ns) generated per query q during training. In our post evaluation experiments we investigated the effects of generating more negative samples, i.e. we regarded all the valid hypernyms over the training set – not being a proper hypernym for q – as h' upon the creation of the (q, h') negative training instances. This latter strategy is referenced as $ns = all$ in Table 36.

	Train		Test	
1A	975(4)	0.41%	1055(4)	0.38%
1B	709(1)	0.14%	767(2)	0.26%
1C	776(2)	0.26%	625(2)	0.32%
2A	442(58)	11.60%	433(67)	13.40%
2B	366(21)	5.43%	341(17)	4.75%

(a) concept

	Train		Test	
1A	379(142)	27.26%	344(99)	22.35%
1B	249(41)	14.14%	205(26)	11.26%
1C	184(38)	17.12%	328(45)	12.06%
2A	0(0)	—	0(0)	—
2B	79(34)	30.09%	102(40)	28.17%

(b) entity

Table 35: Number of in-vocabulary (and out-of-vocabulary, OOV) queries per query type. The ratio of the latter is also shown.

		candidate filtering off						candidate filtering on					
k	ns	MAP	MRR	P@1	P@3	P@5	P@15	MAP	MRR	P@1	P@3	P@5	P@15
200	50	6.5	14.9	13.1	7.4	6.1	5.5	12.1	25.4	18.9	12.9	11.6	10.9
200	all	6.9	15.8	14.1	7.6	6.3	5.8	13.0	27.1	19.9	14.2	12.5	11.8
300	50	6.9	15.8	13.9	7.6	6.4	5.9	12.1	25.7	19.5	13.0	11.5	11.0
300	all	8.0	17.8	15.4	8.9	7.4	6.8	13.5	28.0	21.1	14.5	12.9	12.3
1000	50	9.0	20.0	17.2	9.8	8.3	7.7	13.3	**28.1**	**21.3**	13.8	12.6	12.3
1000	all	11.6	26.1	22.5	12.5	10.8	10.0	**13.6**	27.2	19.4	**13.9**	**13.2**	**12.8**

Table 36: Post evaluation results on the 1A dataset investigating the effect of various hyperparameter choices. k and ns denotes the number of basis vectors and negative samples generated during training per each positive (q, h) pair. Best results obtained for each metric are marked as bold.

		MAP	MRR	P@1	P@3	P@5	P@15
off	off	10.3	21.3	15.0	10.6	10.1	9.6
off	on	10.1	21.1	14.9	10.5	9.9	9.5
on	off	**12.1**	25.4	**18.9**	12.9	**11.6**	10.9
on	on	**12.1**	**25.3**	18.7	**13.0**	**11.6**	**11.0**

Table 37: Ablation experiments, on the 1A dataset with $k = 200, ns = 50$ (and the implementation of `isFreqHyp` fixed). The first two columns indicate whether attributePair$_{ij}$ and FCA-derived features are utilized, respectively.

	MAP	MRR	P@1	P@3	P@5	P@15
1A	76.1	92.2	92.2	82.3	76.4	71.6
1B	71.2	93.4	93.4	78.5	70.9	65.7
1C	81.0	95.9	95.9	87.2	81.7	76.4
2A	72.6	89.6	89.6	81.0	75.3	64.1
2B	95.4	98.8	98.8	97.3	96.0	93.7

Table 38: Test results of an oracle system which uses candidate filtering.

In our official submission we regarded only those hypernyms as potential candidates to rank during test time which occurred at least once as a correct hypernym in the training data. We call this strategy as candidate filtering. Historically, we applied this restriction to speed up the FCA algorithm because this way the size of the concept lattice could be made smaller. As there are valid hypernyms on the test set which never occurred in the training data, our official submission would not be able to obtain a perfect score even in theory. As ceiling analysis, Table 38 contains the best possible metrics on the test set that we could achieve when candidate filtering is applied. In our post evaluation experiments we also investigated the effects of turning this kind of filtering step off. As Table 36 illustrates, however, our scores degrade after turning candidate filtering off.

Our post evaluation experiments in Table 36 suggest that it is advantageous to apply sparse representation of more expressive power (i.e. a higher number of basis vectors). Generating more negative samples also provides some additional performance boost. These previous observations hold irrespective whether candidate filtering is employed or not, however, their effects are more pronounced when hypernym candidates are not filtered.

Finally, we report our post-evaluation results for all the subtasks and compare them to the official scores of the best performing systems in Table 39. It can be seen from these enhanced results for category "all" (concepts and entities mixed) that we would win (1B) Italian and (1C) Spanish. Our post-evaluation system – which only differs from our participating system that it fixes the calculation of a features, does not rely on FCA-based features and uses $k = 1000$ – would also place third in the rest of the subtasks.

5.6.4 *Conclusion*

In this section we experimented with the integration of sparse word representations into the task of hypernymy discovery. We strived to utilize sparse word representations in two ways, i.e. via building concept lattices using formal concept analysis and modeling the hypernymy relation with the help of interaction terms. While our former approach for deriving formal concepts from sparse word representations was not

	MAP	MRR	P@1	P@3	P@10	P@15
1A	13.3	28.1	21.3	13.8	12.6	12.3
1A	19.8	36.1	29.7	21.1	19.0	18.3
1B	**12.5**	24.2	14.5	**13.4**	**12.5**	**12.0**
1B	12.1	**25.1**	**17.6**	12.9	11.7	11.2
1C	**21.8**	**43.8**	**33.7**	**22.9**	**21.4**	**19.9**
1C	20.0	28.3	21.4	20.9	21.0	19.4
2A	21.9	39.5	34.2	25.5	22.6	18.5
2A	34.0	54.6	49.2	40.1	36.8	27.1
2B	31.5	43.6	29.8	30.3	30.3	31.5
2B	41.0	60.9	48.2	44.9	41.3	38.0

Table 39: Post evaluation results for the different subtasks using $k = 1000, ns = 50$ and hypernym candidate filtering. Upper: our system, lower: subtask winner.

successful, the interaction terms derived from sparse word representations proved to be highly beneficial.

Az, ki tőlem elrabolna / Lelkemtől rabolna meg...
'That who stole you from me would rob me of my soul'[17]

— Béni Egressy

6

DEEP CASES

Contents

6.1 OVERVIEW

This chapter, which originally appeared as Makrai (2014) in Hungarian, investigates the argument linking system in a version[1] of `4lang`. As we have already seen in Chapter 3, `4lang` is a multilingual lexicon for general human language understanding containing formal representations of word meaning in the monosemic approach to lexical semantics, which means that items are language independent concepts covering different uses of the same word, uses in different sentence patters and

17 This motto from the libretto of a Hungarian opera is intended here as a Hungarian pun, but we try to explain the joke: Both clauses contain the verbal stem *rabol* 'rob', a pro-dropped syntactic object (an unmarked construction in Hungarian syntax), and an ablative-marked overt argument, but there is a mismatch in the grammatical functions (surface cases):

	(pro-dropped) object	ablative ('from')
el rabolna	Object	maleficient
rabolna meg	maleficient	Object

In the first clause, the ablative is arguably oblique and the preverb *el* 'away' is adverb-like, while in the second clause, the ablative is quirky and the adverbial *meg* is a pure perfectivizer.

1 Since 2014, linking in `4lang` has changed. The last version which is compatible with the present thesis is https://github.com/kornai/4lang/blob/1d19f167b9c0eace5bd874759860781be78f96ed/4lang. For a more minimalistic treatment of linking that relies only on `AGT` and `PAT` see Kornai (2022, Ch 2.4.).

even in different parts of speech with the same meaning representation.[2] Multilinguality and abstractness of items have the effect that a simple deep case (or thematic) frame captures uses with different arity (i.e. transitive and intransitive). Deep cases denote the nodes in the graph representing the meaning of a predicate where the representation of the argument (single word, entity or phrase) has to be inserted.

`4lang` makes no clear cut between complements and adjuncts. Basically an argument is represented by a deep case whenever its needed for building the representation of the verb. As uses of the same verb with different arities are handled in the same item, deep cases are used consequently in different verb patterns, and all possible arguments are included in the representation. However, as verbs can be defined as special cases of other verbs (biting is cutting with teeth), arguments are inherited, so not every argument is listed directly in the definition of some verb. Another source of implicit arguments are constructions providing verbs with outer arguments e.g. *paint a picture* ***for somebody***, *sleep* ***an hour***, *fly* ***the Atlantic***. Causatives (e.g. *march* ***the soldiers***) are also attributed to constructions rather than argument structure.

Most frequent verbal deep cases are agents (denoted by `AGT`), patients (`PAT`), and datives (`DAT`). Patient plays the role of the neutral case it seems to play in many systems (Somers 1987). Following the unaccusative hypothesis, arguments of intransitive verbs split to agents and patients. The label "dative" is taken from Fillmore (1968), but our understanding is narrower as we mainly restrict dative to recipients in ditransitives (verbs of communication (e.g. *tell*) and transfer (e.g. *give*)). These verbs correspond to Schank's (see Section 2.1.3) transmission, MTRANS and PTRANS. There are three locative cases in `4lang` (`TO`, `FROM`, and `AT`), the latter being used for the abstract goal of relational nouns such as *occasion* and *need* as well. A greater group of relational nouns require the possessive (`POSS`) such as *absence* and *duty*. Quirky cases can be marked in a language dependent module.

Deep cases in `4lang` are not restricted to verbs. Some grammatical features such as plural contribute to meaning. Thechnically, the definition of these morphemes refer to the referent with `REL`. Representations of productive derivational suffixes and adpositions also refer to the conceptual element they attach to with `REL`.

To calculate the meaning representation of a sentence, we need to map the predicate-argument relationships. From a theoretical linguistic point of view, we have two pillars here: selection constraints and surface cases in the broadest sense (e.g. the order of phrases, case affixes and/or adpositions varying from language to language). In the opinion of our research group, selection constraints correspond to *spreading activation*

2 The lexicon, automatically collected word forms in 50 languages, a vector space language model (embedding) computed from `4lang`, and articles can be found at http://hlt.sztaki.hu/resources/4lang/

in the dictionary, and the knowledge about surface cases is indirectly encoded by deep cases. From the point of view of deep cases, it is important that `4lang` is designed to connect to each language with a language-specific module, which tells which surface cases will realize each deep case in that language. In this chapter we deal with deep cases, so we outline the activation spreading only briefly and in a simplified way.

Recall the definition graph, the vertices of which are concepts in the dictionary, and two of these are connected if one is included in the definition of the other (Section 3.3), e.g. 'milk' is associated with 'liquid'. If we want to know which argument of *drink* the word *milk* fills in a sentence, we look for the shortest path (edge sequence) between the two concepts in the graph. With some luck, this passes through the word *liquid* and largely corresponds to the representation of the phrase *drink milk*.

Let us now turn to how it can be calculated from surface cases which role each argument plays (with slot they fill). The meaning representation of a term that includes an predicate with its arguments (e.g. a verb phrase) should be calculated from the following: the representation of the meaning of the predicate, that of the arguments, and the structure of all these together. In the case of `4lang`, the latter is taken care of by indicating in the meaning representation of the predicate (typically a verb) where the meaning representation of each argument should go. To do this, we need to be able to distinguish the arguments of higher arity predicates (e.g. transitive verbs). This is done with reference to the deep case of the argument. The background for our method is the common assumption that, at least within languages, there are regular correspondences between the semantic role (e.g. agent) and the syntactic properties of the arguments (the surface case of the argument, which sentence alternations the verb participates in), and in several cases these regularities shows up in more languages.

As we have already written, our deep cases only serve to identify which argument is which. In this context, it is perhaps worth emphasizing that the classification of arguments into deep cases is not primarily a semantic classification. In computer semantics, the fact that there is a regular difference between the meanings of the corresponding arguments could often be an argument in favor of distinguishing between two deep cases. For example, Talmy attributes the intentional difference between the verb pairs *hide/mislay, pour/spill, ...* to the exact nature of case of the agent.

In Allen and Teng (2018)'s view, semantic roles should have consequences independent of the predicate or event. They explore three aspects: entailment from a role independent of the type that has such roles; integration with ontology (Roles should obey the typical entailments in an ontology, e.g. inheritance of properties from parents); and derivability (roles should be derivable from the definitions in dictionar-

ies). Theses authors admit that only the third property allows empirical evaluation. In `4lang` such differences do not justify the introduction of a new deep case, as the meaning is fully described in the definition field of the lexical entry.

Compared to semantic-based classification, the other extreme is where the number of cases cannot exceed the largest number of arguments we encounter among verbs. We do not strive for this either, as we want to take advantage of regularities between the semantic role and the syntactic properties.

6.2 INDIVIDUAL RELATIONS

6.2.1 *Function morphemes*

How does `4lang` grasp simpler dependencies? On the one hand, certain inflections, such as the plural, have a conceptual meaning in the sense that in the representation of the structure containing the inflectional affix, there is an element for which the inflectional affix is responsible. Productive derivational affixes and adpositions are similar. We need to treat these relations (stem–inflectional affix, stem–derivational affix, adpositional object–adposition) uniformly already because `4lang` wants to be language-independent, and the same semantic relation is expressed differently in different languages, e.g. the meaning, which is expressed by the possessive personal suffix in Hungarian, is expressed by the possessive pronoun in English. Here, the place of the representation of a function morphemes in the representation of the more content element is always represented by the keyword `REL` (*relational, related*), which in a broader sense can be called a deep case.

6.2.2 *Verbal deep cases*

6.2.2.1 *Argument positions, alternations, open case inventory*

Turning now to the arguments of verbs, we must first clarify what we mean by argument. Only the obligatory ones or the adjuncts as well?[3] Are we talking about surface arguments, or the arguments for the (deep, logical) predicate corresponding to the verb in a formal semantic translation? In the first approximation, we follow the literature (Somers 1987) in representing those surface arguments by their deep case in the definition of a verb that are needed to describe the meaning. Another issue arises from the fact that, due to the abstract nature of `4lang`, we do not differentiate between the transitive use of a verb (or even that with more surface arguments) and the intransitive use of the same verb

3 This chapter was originally published in Hungarian, where there is a common term for arguments and adjuncts, *bővítmény* ‘expansion’, arguments proper are called *vonzat* ‘attractee’, and adjuncts proper are called *szabad bővítmény* ‘free expansion’.

AGT	383
PAT	311
REL	81
POSS	52
DAT	30
TO	17
FROM	11
AT	2

Table 40: Each deep case with the number of predicates using them

form. Deep cases are defined in such a way that the same predicate in different uses gets the same case. It follows that if a verb has a transitive use, the deep case of two participants must also be indicated. Finally, a further nuance is that when a verb can be defined as a special case of another verb and the arguments are inherited, it is not necessary to explicate them in the definition, e.g. *bite* is defined as `cut, INSTRUMENT tooth` ('cut with tooth'), and *bite* inherits the arguments of *cut*, so these are not listed.

In choosing deep cases, it is not our task to create harmony between the participants of different verb roots. Thus, for example, it is not our intention that the participants in the sentences *John sells a book to Peter* and *Peter buys a book from John* will receive the same deep cases for the two sentences.[4] Finally, we do not include *outer roles* in the verb definition, that is, the possible extensions that can be assigned to a verb by a construction that affects entire verb classes (e.g. motion verbs) or even all verbs, so in the following examples the putative argument position corresponding to the bold face phrases: *paint an image* ***to someone****, sleep* ***an hour****, fly* ***the Atlantic Ocean*** *over*.[5] Causation[6] is also considered such construction.

6.2.2.2 *Individual cases*

There are 744 verbs in `4lang`. Deep cases are listed in the Table 40, along with the number of words that they occur with. Unsurprisingly, the most common deep case is the agent. When writing definitions, we can decide without much difficulty which argument of a typical[7] transitive verb is the agent (indicated by the keyword `AGT` in the dictionary). The second most common deep case in `4lang`, which we called patient

4 In both cases, the English subject will be an agent, and the subject will be `PAT`.
5 For more on external roles, see Somers 1987, Chapter 9.
6 In Hungarian, causation is marked by a derivational suffix *-(t)At*.
7 Psych verbs are more problematic, but there are so few of them in the defining vocabulary, that we can treat this class as exceptional (quirky).

	object-marking	ergative 1	ergative 2	active	lexicalized active	subject-marking
Peter is writing the letter.	nom	ag	ag	ag	ag	ag
Peter is writing.	nom	nom	ag	ag	ag	ag
Peter is walking.	nom	nom	nom	ag	ag/nom	ag
Peter is ill.	nom	nom	nom	nom	ag/nom	ag

object marking	English (eng), Hungarian (hun)
ergative 1	Kabardian (kgb), Avar (ava), Adige (ady)
ergative 2	Aghul (agx), Udi (udi)
active	Bats (bbl)
lexicalized active	Georgian (kat), Dakota (dak)
subject marker	Mingrelian (xmf), maidu (nmu)

Table 41: Arguments of intransitive verbs in different languages (Komlósy 1982). The SIL code of the languages is also indicated.

(PAT), is often defined only as the "semantically unmarked" deep case, but since the others are relatively clearly identifiable, this is not a problem either. According to the unaccusative hypothesis used in modern syntax, the argument of an intransitive verbs can also be patient (e.g. *fall, melt*).

Komlósy (1982) gives a good summary of how the agent and patient of intransitive and transitive verbs are classified by surface cases in different languages. Komlósy reviews a number of ergative (or active and subject-marking) languages in terms of the case of the arguments of different single-argument verbs. Table 41 shows where the different languages draw the line between the two cases on a scale of activity. These data suggest that in a language-independent case system we need to make finer differences than the binary AGT vs PAT partition. It is a question of whether this would really improve the performance of our systems in these languages. Such experiments would exceed the bounds of the present thesis, so we'll stick with the simpler case set.

With agent and patient, we essentially follow the generative semantic tradition. We deviate more from the history by using the *dative* (DAT). The name is taken from the oldest terminology of generative semantics (Heringer 1967; Fillmore 1968). Fillmore himself later separated the dative into experiencers, objects (*Object*), and goals (*Goal*). We basically use the dative only for verbs with at least three surface arguments, in other cases only based on their similarity to the former. As for their meaning, some of the three-argument verbs are the special cases of *say*, very reminiscent of Schank's (see Section 2.1.3) mental transmission MTRANS: *(admit, allow, command, declare, emphasize, explain, express, forbid, grateful, say, swear, teach, thank)*. Another group is related to *give*, i.e. Schank's physical transmission PTRANS: *(bestow, have, help, lend, let, make offering, offer, owe, owing to, pass, pay, present, sell,*

show). We note that though the predicative complement of verbs like *appear, regard*, and *seem*, are marked dative in Hungarian, they are not dative: in a division finer than ours (Chafe 1970), complement would be a case on its own, which we classify as `PAT`. There are some further words in the defining vocabulary with dative marked arguments in Hungarian (*nehéz* 'difficult; heavy', *y* tetszik *x*-`dat` '*x* likes *y*') or German (*ähneln* 'resemble', *beitreten* 'join', *gleichen* 'equal'), but these are too sporadic to draw any generalization, so we treat them as exceptional.

There are as many as three *locative* cases in `4lang`, `TO` and `FROM` corresponding to the Fillmore Goal (*Goal*) and Source (*Source*), and the essive `AT`. We have gone as far as possible in abstraction: if an argument in many languages gets a surface case that is also used to express the goal of movement (specific inflectional suffixes in Hungarian, and prepositions in English), then we consider it a goal. We mean *able, accustom, add, addition, available, belong, gentle (hu:gyengéd, la:mollis, pl:delikatny), include, invite, join, law, listen, load, mix, necessary, need, occasion, put, ready, remind, sensitive, similar, skill, tendency.* The other two locative cases are the source (*accept, borrow, buy, cut off, date, derive, of, profit, remove, rent, rubber, separate, subtract, take*) and the essive space (*situated, stay*).

In the language-specific module already mentioned it is possible to mark some arguments of some verbs with surface cases, if their case is unpredictable from their deep case (*quirky case*). On the other hand, it is already clear from English, Hungarian and German that there are verbs where no generalization seems useful. In this case, we use the same `REL` keyword as for predicates with a single surface argument, e.g. *prefer to something.*

6.2.3 *Relational nouns*

Finally, consider the relationship between *relational nouns* and the word associated with them (e.g. in the case of *interest*, the stakeholder). The phenomenon that makes the noun *interest* relational is twofold. On the surface hand, the proportion of possessed occurrences of the word *interest* is significantly higher than among other nouns. On the other hand, which is more interesting from a semantic point of view, no matter how we want to describe the meaning of the word *interest*, we would probably refer to the "stakeholder". The grammatical relationship between the two words is possessive in most relational nouns, but we find something different in about one-tenth of the lexemes. In the case of the words *occasion* and *need*, the participant which we call the goal for lack of a better word, is sublative in Hungarian (*-rA*, lit. onto) and *for* in English. In the representation of relational nouns, we use the keywords `POSS` or `TO` according to the grammatical relationship between the two words to indicate the place where the representation of the related word (the interested person and the target, respectively) goes. `TO`

is the same abstract goal we encountered at verbs. By mediating deep cases, the linguistic relationship thus helps to find the semantic relationship between the two things (the interested and the interest, and the occasion and the goal). We will not handle relational nouns that are productively formed from a verb, because `4lang` does not distinguish e.g. participles from the corresponding verb.

6.3 CONCLUSION

We have shown how deep cases can work in a freely available machine comprehension resource that assigns deep cases directly to rather abstract language-independent concepts in each language. Future research could test the operation of deep cases in a machine comprehension task. Another topic of future research may be how to inherit semantic arguments of verbs in a broader vocabulary, possibly using the human language definition of the words in dictionaries that define them. Finally, the long-term goal involves generation. In the even longer run, systems should be able, give some paraphrase, select the one which is the best suited to some pragmatical situation s.

Keywords: language resource, syntactic analysis, verb structures, Mazsola, size

— Bálint Sass (2015)

7

DECOMPOSING A TRANSITIVE VERB TENSOR

Contents

7.1 INTRODUCTION

Verbs have been characterized on the basis of how frequently various syntactic constituents occur in various grammatical relations to them, which is, not surprisingly, related to the meaning of the verb (Levin 1993). These selectional preferences have been analyzed with machine learning tools (Van de Cruys 2009). Verb structures include collocations, whose syntactic modifiability or semantic compositionality is reduced: their linguistic distribution may be idiosyncratic or the sense of the combination may be habitual or even fixed (Bouma 2009).

Tensors (>2-dimensional arrays) generalize matrices; while matrices contain numbers aligned in two dimensions, rows and columns, tensors have more of these dimensions, also called *axes* or *modes*[1] Singular value decomposition (SVD) of a co-occurrence matrix is a natural tool to compute generalizations about the interactions between two modes, like words and documents (LSA, Landauer and Dumais (1997), Section 4.1.3), target and context words (words embeddings, Mikolov, Sutskever, et al. (2013), Levy and Goldberg (2014c), and Pennington, Socher, and Manning (2014)), or words and dependency contexts (Levy and Goldberg 2014a). Four ways of looking at SVD (in LSA) can be distinguished (Turney and Pantel 2010): the goal can be the modeling of some latent meaning, noise reduction, indirect aka high-order co-occurrences (when two words appear in similar contexts), or sparsity reduction. Intuitively, language features multi-mode interactions: *the turntable playing the piano* can be strange (Van de Cruys 2009), while the two-mode relations ⟨play, SUBJ, turntable⟩ and ⟨play, OBJ, piano⟩ are perfect. Tensor generalizations of matrix decomposition (Kolda and Bader 2009), especially *low-rank factorizations*, open the way for the analysis of such interactions.

It seems that, after intensive early research (Van de Cruys 2009; Van de Cruys, Poibeau, and Korhonen 2013; Polajnar, Rimell, and Clark 2014; Fried, Polajnar, and Clark 2015; Hashimoto and Tsuruoka 2015), results obtained with skip-gram and related word embedding methods outshone tensor methods for verb argument structure. Yet tensor decomposition has developed remarkably, and NLP test-beds in the domain of verb argument structure have been involved in cutting-edge scalable, noise-robust tensor works (Sharan and Valiant 2017; Bailey, Meyer, and Aeron 2018; Frandsen and Ge 2019). The data-driven linguistic understanding of word ambiguity and especially that of verb selection is still immature. Here we try to make progress in the linguistic direction by further research on tensorial analysis of verb argument structure.

Tensor decomposition provides embedding vectors for each mode (in our case, nouns as subjects, verb, and nouns as objects) analogous to word embeddings in (shallow or deep) neural networks. In this paper, we compute different association measures between subjects, verbs, and objects, populate tensors with these measures, decompose the tensors with different algorithms, and investigate the resulting word embeddings quantitatively and qualitatively to answer the following questions.

Our first four questions will be answered quantitatively in the modeling of English subject-verb-object triple similarity, while the last two questions are qualitative.

- Which *association measure* yields the best representations? We experiment with several measures, including our novel generaliza-

1 The term *mode* is preferred when data from different modalities are fused.

tion of normalized pointwise mutual information to the higher-order (>2) case.

- Should we include *empty argument fillers* (subjects or objects) in our co-occurrence statistics? Ideally, including them may help generalization over the transitive and the intransitive uses of the same verb, while discarding them may help focusing on transitive structures cleanly as a separate phenomenon.

- The two tensor decomposition algorithms, CPD and Tucker, which we will introduce in Section 7.3, have very different time-complexity: Tucker is much faster. Tensor decomposition has hyper-parameters like the decomposition rank and the frequency cutoff. Both are related to memory limitation, especially the latter. It would be beneficial, *if the two algorithms reached the best results with the same hyper-parameters,* because then a fast parameter tuning with Tucker would also benefit CPD. Is this the case?

- How does the trade-off between the three hyper-parameters related to the *size of the decomposition* (i.e. the decomposition rank, the inclusion of empty fillers, and the frequency cutoff) look like?

- Do latent dimension of our word embeddings reflect lexical knowledge?

- Can the difference between each noun as a subject versus an object correspond to some intuitive difference between subjecthood and objecthood?

Section 7.2 describes the linguistically motivated association measures between subjects, verbs, and objects we apply. These measures include ones that are novel to the best of our knowledge. Section 7.3 offers an introduction to tensor decomposition. Section 7.4, most of which originally appeared in Hungarian as Makrai (2020), reviews the computational linguistic applications of tensor decomposition, especially those related to verb argument structure. Last but not least, Sections 7.5 to 7.7 describe our experiments. Our code is available online.[2]

7.2 COUNTS, WEIGHTING, AND ASSOCIATIONS

Word co-occurrences form *sparse* arrays, as most words do not occur empirically with most words, and frequencies span many orders of magnitude (*Zipf* or power law distribution, Manin (2008) and Gittens, Achlioptas, and Mahoney (2017)). In order to scale to large data, linguistic tensor decomposition methods have to be based on sparse tensors populated with more sophisticated scores than frequency. Now

2 https://github.com/makrai/verb-tensor

we turn to these weighting functions and especially to linguistically motivated association scores.

The simplest choice is the logarithm of the co-occurrence frequency (Pennington, Socher, and Manning 2014; Sharan and Valiant 2017). Jenatton et al. (2012) places the modeling of the $\langle \text{subject}, \text{verb}, \text{object} \rangle$ triples in the context of multi-relational learning, and apply a weighting function related to the log-bilinear model (Mnih and G. Hinton 2007; Mikolov, Chen, et al. 2013).

Van de Cruys (2009, 2011) and Van de Cruys, Poibeau, and Korhonen (2013), and Bailey, Meyer, and Aeron (2018) use three-mode generalizations of the information-theoretic association measure *(Positive) Pointwise Mutual Information* ((P)PMI). Positivity is related to sparse inputs: in order to attribute higher scores to actual co-occurrences than unattested ones, PMI and the lexicographic association scores introduced in the following paragraph, *positive* variants of the association measures have to be used, e.g. PPMI, which replaces negative PMI entries with zero. We discuss the two types of three-variable generalization of PPMI in Section 7.2.2: the more standard total correlation (that we still call PMI) and interaction information.

We also experiment with generalizing Log Dice (Rychlý 2008) to three axes

$$\log \frac{3f(x,y,z)}{f(x)+f(y)+f(z)} + c,$$

where c is chosen so that the Log-Dice values are non-negative. (While 3 in the nominator is redundant, because it is subsumed under c, we keep it in the formula to make it more reminiscent of the established 2-variable case.) The use of Log Dice as well as salience introduced in the next paragraph has, to the best of our knowledge, mainly been limited so far to lexicography.

7.2.1 *Salience and normalized PPMI*

PPMI, despite of its nice information-theoretic interpretability, is biased towards rare events (Turney and Pantel 2010; Levy et al. 2015; Zhuang et al. 2018). This motivates the Sketch Engine lexicographic software (Kilgarriff et al. 2004) to multiply vanilla PPMI by $\log f$ (in our case, by $\log(f(x,y,z))$, to get the measure of *salience*. We apply similar modifications to every score introduced in Section 7.2 so far. Denoting vanilla PPMI, interaction information and Log Dice by `pmi-vanl`, `iact-vanl`, and `Dice-vanl`, respectively, we get `pmi-sali`, `iact-sali`, and `Dice-sali` by multiplying the vanilla score by $\log f(x,y,z)$.

There is a theoretically better motivated way of transforming PMI to some measure which is less biased towards rare combinations. In Bouma (2009)'s approach, *normalization* is related to boundedness. He looks for measures whose absolute value is pointwise larger than that of PMI. Entropy and negative log probability are two of those measures,

and we follow the literature in choosing the latter. In our experiments, we apply this normalization to the two multi-mode generalizations of PMI which will be introduced in Section 7.2.2, interaction information and the one which we will still call PMI. While normalized interaction information does not excel in our experiments, tree-variable normalized PMI, which is to the best of our knowledge the novelty of the present paper, proves the best among the alternatives considered. Empirically, when divided by $-\log p(x, y, z)$, positive interaction information and the more standard 3-mode PPMI is upper-bounded by 1 and 2, respectively.

7.2.2 *Higher-order PMI*

One would think that it's obvious that the 3-variable generalization of Pointwise Mutual Information (PMI) is

$$\log \frac{p(x, y, z)}{p(x)p(y)p(z)}, \tag{7}$$

but it turns out that this is only one of the possible generalizations. Van de Cruys (2011) introduces two pointwise association measures, whose expected values are two different multivariate generalizations of mutual information (Shannon and Weaver 1949): interaction information (McGill 1954) and total correlation (Watanabe 1960).

Pointwise *interaction information* is based on the notion of conditional mutual information.[3]

$$\log \frac{p(x, y)p(x, z)p(y, z)}{p(x, y, z)p(x)p(y)p(z)}$$

Total correlation on the other hand quantifies the amount of information that is shared among the variables, with a pointwise variant defined by the formula in Equation (7). Following the literature (Villada Moirón 2005; Van de Cruys 2009; Van de Cruys, Poibeau, and Korhonen 2013; Bailey, Meyer, and Aeron 2018), when we speak about *(multivariate Positive) Pointwise Mutual Information* in this paper, we will mean (pointwise) total correlation.

Van de Cruys (2011) reports that in their Dutch experiments both methods are able to extract salient subject verb object triples (prototypical SVO combinations like *poll represents opinion* and fixed expressions). Narrowing the scope to the word *play*, they find that interaction information picks up on prototypical SVO combos e.g. *orchestra plays symphony*, while the more established one (which he calls specific correlation) picks up on *play a role* and salient subjects that go with the expression.

3 Mnemonically, the formula of the pointwise variant generalizes the 2-mode case along the inclusion and exclusion principle, except it has the numerator and the denominator swapped to ensure a proper set-theoretic measure.

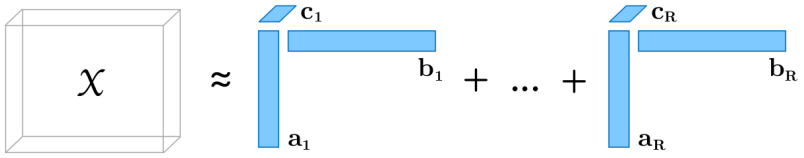

Figure 16: Canonical Polyadic Decomposition, figure from Rabanser et al. (2017).

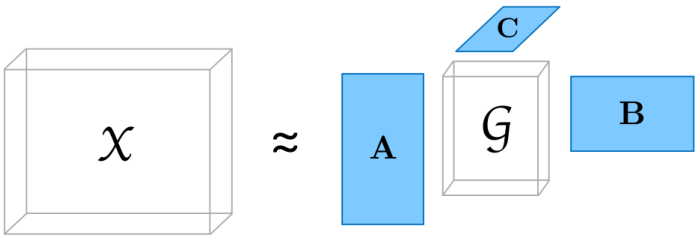

Figure 17: Tucker Decomposition, figure from Rabanser et al. (2017).

7.3 TENSOR DECOMPOSITION

The main entry point to tensor computation is Kolda and Bader (2009), but Rabanser, Shchur, and Günnemann (2017) is also worth consulting.

There is no single generalization of the SVD concept, the two most popular extensions, Canonical Polyadic Decomposition and the more general Tucker, feature different generalized properties. Sidiropoulos et al. (2017) discuss the interpretation of these two different ways of decomposition in signal processing and machine learning points of view.

7.3.1 *Canonical Polyadic Decomposition*

Canonical Polyadic Decomposition (CPD, aka CanDecomp, `Para`llel `Fac`tor model, CanDecomp, rank decomposition, or Kruskal decomposition, (Carroll and Chang 1970)) expresses a tensor as a minimum-length linear combination of rank-1 tensors. A rank-1 tensors is the tensor product of a collection of vectors, just as the dyadic product of two vectors is a 1-rank matrix, see Figure 16.

The alternating least squares algorithm (ALS, Carroll and Chang (1970) and Harshman (1970)) is an iterative method for CPD. In each iteration, all but one of the modes are fixed and the remaining one is fitted. ALS does not guarantee convergence, and even if it converges, this cannot be detected in a trivial way. Orth-ALS (Sharan and Valiant 2017) improves on ALS.

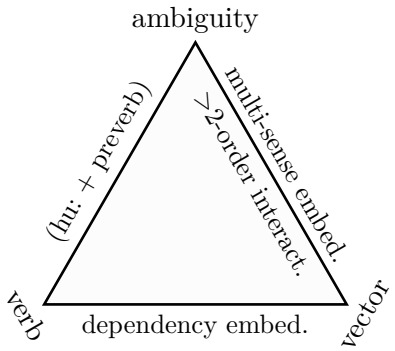

Figure 18: Three related topics. *"Not in the unity of a single person, but in a Trinity of one substance."*

7.3.2 *Tucker decomposition*

While CPD seems more relevant for linguistics representation, we also discuss Tucker decomposition, because it can be computed much more efficiently. Tucker decomposition (aka Higher Order SVD, Tucker (1966)) factorizes a tensor into a core tensor $\mathcal{G}$ multiplied by a matrix along each mode, see Figure 17. In the case of

$$\text{subject} \times \text{verb} \times \text{object}$$

tensors, rows of the three matrices contain embedding vectors of entities (subjects or objects) and those of verbs ("relation"), and entries of the core tensor $\mathcal{G}$ determine the levels of interactions between the former three. Tucker decomposition is not unique, because we can transform $\mathcal{G}$ without affecting the fit if we apply the inverse of that transformation to the factor matrices. Uniqueness can be improved (Kolda and Bader 2009) by imposing e.g. sparsity, making the elements small, or making the core "all-orthogonal". Other priors and constraints in tensor learning involve non-negativity and independence (Lahat, Adali, and Jutten 2015).

7.4 RELATED WORK

7.4.1 *Ambiguity, verbs and vectors*

Figure 18 illustrates various relations between ambiguity, verbs, and vectors. Word sense disambiguation (WSD, Pilehvar and Collier (2016), Collados et al. (2016), and Alexander Panchenko et al. (2016)) and induction (WSI) are the tasks of classifying or clustering word tokens to senses (with or without supervision), respectively.

Verbs offer a field where ambiguity interacts with argument structure (selectional restrictions, Vulić, Mrkšić, and Korhonen (2017), Majewska et al. (2018), and Sun et al. (2010)).

	corpus	shape	weighting, postprocesssing	rank
Van de Cruys (2011)	Dutch .5 B	10 K subjects × 1 K verbs × 10 K direct objects	PPMI	50 … 300
Van de Cruys (2011)	Dutch .5 B	10 K subjects × 1 K verbs × 10 K direct objects	2 variants of PMI	(no decomp)
Van de Cruys, Poibeau, and Korhonen (2013)	UKWaC 2 B	10 K subjects × 1 K verbs × 10 K objects	PMI	300
Jenatton et al. (2012)	2 M Wp articles	30 K subjects × 5 K verbs × 30 K direct objects	$\mathbb{P} = 1/(1 + \exp(-\mathbf{s}_i \cdot \mathbf{R}_j \otimes \mathbf{o}_k))$ + refinement	25, 50, 100
Sharan and Valiant (2017)	Wikipedia 1.5 B	10 K words × 10 K words × 10 K words	$\log(f + 1)$, $w_i = s_i \oplus v_i \oplus o_i$ normalized	100
Bailey, Meyer, and Aeron (2018)	.3 B from Wp	word freq cut-off = 1 000	($\pm$shifted) PPMI, w_i normalized	300

Table 42: NLP-oriented tensor decomposition work. Corpus sizes are shown in billion words. In the formulae, f denotes co-occurrence frequency.

Multi-sense word embedding models (Reisinger and Mooney 2010; Huang et al. 2012; Neelakantan et al. 2014; Bartunov et al. 2016; Li and Jurafsky 2015; Borbély, Makrai, et al. 2016; Makrai and Lipp 2018) model different meanings of word forms with different vectors in unsupervised fashion.

Dependency-based word embeddings (Levy and Goldberg 2014b; MacAvaney and Zeldes 2018) on the other hand use syntactic relations provided by dependency parsers as contexts for target words. Amrami and Goldberg (2018) apply deep word representations (Peters, Neumann, Iyyer, et al. 2018; Devlin et al. 2018; Howard and Ruder 2018) to achieve state-of-the-art for unsupervised WSI. Peters, Neumann, Zettlemoyer, et al. (2018) analyses the WSD information (among other features) captured in the representations demonstrating that a language modeling on its own provides WSD performance close to the state-of-the-art.

Tucker decomposition also proved promising for link prediction aka. Knowledge Graph Completion, where elements of the tensor encode facts: in the input tensor, 1 indicates a true fact, and 0 indicates unknown (false or a missing). The task is to suggest missing true values among the 0s. Kazemi and Poole (2018) and Balažević, Allen, and Hospedales (2019) offer solutions for the problem that Tucker decomposition performs poorly for this task, as it learns two independent embedding vectors for each entity, a subject and an object vector. In Balažević, Allen, and Hospedales, subject and object entity embedding matrices are assumed equivalent.

7.4.2 *Tensors for language*

Table 42 summarizes some features of NLP-oriented related work.

Van de Cruys (2009) introduces a non-negative tensor factorization model for selectional preference induction. Van de Cruys, Poibeau, and Korhonen (2013) develop this line of research by concentrating on compositionality and modifying the tensor factorization model to the minimization of the Kullback-Leibler divergence, which fits better to long-tail distributions we find in language. The latent models (i.e. word vectors) for nouns are fixed to values computed from standard co-occurrence data, and the induction of three-way subject-verb-object interactions is inspired by Tucker decomposition. Fixing subject and object vectors to two-order co-occurrence data limits the exploitation of three-order structure provided by tensors.

Jenatton et al. (2012) learn semantic verb representations in the context of multi-relational learning that originally involves data-sets describing multiple relations (now verbs) between entities (now nouns), e.g. social networks, recommender systems, the semantic web, or bioinformatics data. In this paradigm, a collection of relations is modeled, where the relations themselves can be similar in some respect to each other. In their experiments, entities have a unique representation shared

between relation types. The linguistic tensor is trained on a corpus of ⟨subject, verb, direct object⟩ sentences.

Zhang et al. (2014) investigate how manually created semantic resources can be combined with neural word embeddings to separate synonyms from antonyms, relations that are notoriously difficult to distinguish with distributional means. They inject the thesaurus data as the first slice (pane) of their tensor and the distributional similarities as the second one.

Sharan and Valiant (2017)[4] compute a generic word embedding from a symmetric 3-mode tensors with *Orthogonal-ALS*, a modification of the ALS approach that is as efficient as standard ALS, but provably recovers the true factors with random initialization under standard incoherence assumptions on the factors i.e. that the factors have small correlation with each other, what is satisfied in NLP problems, where the rank of the recovered tensor is typically significantly sublinear in the dimensionality of the space. Orthogonal-ALS periodically "orthogonalizes" the estimates of the factors, thus preventing multiple recovered factors from "chasing after" the same factors. They get the word embedding by concatenating the three recovered factor matrices (with 100 latent dimensions each) into one matrix (with 300 columns) and normalizing the word vectors. These authors do not evaluate their results in respect to >2-order relationships in NLP.

Sharan and Valiant (2017) evaluate their embeddings obtained as an orthogonalized tensor in the standard word analogy (i.e. "puppy is to dog as kitten is to x") and semantic word-similarity tasks. The use of Orth-ALS rather than standard ALS leads to significant improvement, but the matrix SVD method still outperforms the tensor based methods. After considering the pessimistic option that natural language may „not contain sufficiently rich higher-order dependencies among words that appear close together, beyond the 2-mode structure", they give another possible explanation that the two tasks they evaluated on may not require this higher (>2) order statistics.

Zhuang et al. (2018) propose to use second-order co-occurrence relations to train word embeddings via a newly designed metric.

While it was rejected from ICLR 2018, we also mention Bailey, Meyer, and Aeron (2018)[5], who train a 3-mode super-symmetric tensor that is remarkable from the word sense induction perspective: it turns out that representations for each meaning of a polysemous word is obtained by multiplication with an appropriate context vector. Bailey, Meyer, and Aeron also mention the relation between learning tensors and Gaussian mixture models (GMM), specifically GMMs that capture polysemy in word embeddings (Athiwaratkun, Wilson, and Anandkumar 2018; Anandkumar et al. 2014). We leave this to future work as we do not enough experience with GMMs to go in more details here. Bailey

4 http://web.stanford.edu/~vsharan/orth-als.html
5 https://github.com/popcorncolonel/tensor_decomp_embedding

⟨athlete, run, race⟩	finish (.29), attend (.27), win (.25)
⟨user, run, command⟩	execute (.42), modify (.40), invoke (.39)
⟨man, damage, car⟩	crash (.43), drive (.35), ride (.35)
⟨car, damage, man⟩	scare (.26), kill (.23), hurt (.23)

Table 43: Most similar verbs to verbs contextualized in transitive structures (Van de Cruys, Poibeau, and Korhonen 2013).

et al. also emphasize the importance of analyzing the performance as a function of training set size (Jastrzebski et al. 2017), which is commonly done in transfer learning evaluation.

Frandsen and Ge (2019)'s model captures specific syntactic relations between words with correlations between three words (measured by their PMI) form a tensor.

7.4.3 *Evaluation in related work*

7.4.3.1 *Qualitative analysis*

Van de Cruys (2009) finds that among the 100 dimensions they train, 44 exemplify frame semantics. In a dimension we could call *police arrest suspect*, subjects, verbs and objects with the greatest weight are words like *police, arrest*, and *suspect*, respectively. Other examples are *majority support proposal* or *government send troop*. The semantics of another 43 dimensions is less clear: they represent single verbs, or different senses of a verb get mixed up. Thirteen dimensions are based on fixed expressions e.g. *x play role*, the subject slot being distributed evenly among dozens of words, e.g. *revenge, shame, institution*, or *culture*.

In the tensor by Van de Cruys, Poibeau, and Korhonen (2013), slices represent verbs. They illustrate the data by showing the most similar verbs to query verbs contextualized in triples, see Table 43.

In the qualitative part of their evaluation, Frandsen and Ge (2019) search for the words with closest embedding to the composed adjective-noun and verb-object vectors.

7.4.3.2 *Quantitative analysis*

Van de Cruys (2009) evaluates his model in pseudo-disambiguation where the task is to judge which subject (s or s') and direct object (o or o') is more likely for a particular verb v. The test set is constructed by drawing $\langle s, v, o\rangle$ from the corpus, while s' and o' are a subject and a direct object randomly chosen from the corpus, e.g. *youngster/coalition drink beer/share*.

They evaluate their system in the similarity task for transitive sentences (Grefenstette and Sadrzadeh 2011), which is an extension of the similarity task for compositional models (Mitchell and Lapata 2008).

Jenatton et al. (2012) evaluate their model in two tasks: verb prediction given a subject and a direct object, and lexical similarity classification. They observe that the latent representations are sparse or, more precisely, dominated by few large values: the top 2% of the largest values account for about 25% of the ℓ_1 norm.

Zhang et al. (2014) evaluate their work in antonym questions (Mohammad, Dorr, and Hirst 2008).

Bailey, Meyer, and Aeron (2018) evaluates in two groups of tasks: one includes a modification of outlier detection (Camacho-Collados and Navigli 2016) and some supervised tasks, and the other consists of POS classification without sentential context, sentiment analysis (as described in Schnabel et al. (2015), which in turn heavily builds on Maas et al. (2011)), and word similarity.

Frandsen and Ge (2019) evaluate their work in the adjective-noun phrase similarity task (Mitchell and Lapata 2010).

7.4.4 *Hungarian*

WSD for Hungarian in the machine-learning sense goes back at least to Miháltz (2005) and Vincze et al. (2008). Verbs have been in the focus of researchers ranging from corpus linguists to NLP proper (Dressler and Ladányi 2000; Kuti, Héja, and Sass 2010; Miháltz and Sass 2013). The main databases of verb constructions are Mazsola (Bálint Sass 2015; Bálint Sass 2018), Tádé (Kornai, Nemeskey, and Recski 2016), and Manócska (Kalivoda, Vadász, and Indig 2018; Ágnes Kalivoda 2019). Word embeddings of Hungarian have been analyzed by researchers including Makrai (2016a), Siklósi (2016), and Szántó, Vincze, and Farkas (2017)[6].

7.5 EXPERIMENTS

In this section, we report our experiments. After introducing in Section 7.5.1 the corpus that serves as the basis of our empirical investigations, Section 7.5.2 compares association measures, the two alternatives for treating missing arguments, the two decomposition algorithms, and some other hyper-parameters (the decomposition rank and the frequency cutoff) in the classical task of predicting the similarity of English subject-verb-object triples (Kartsaklis and Sadrzadeh 2014). Then in Section 7.5.3, we investigate the latent dimensions qualitatively. Section 7.5.4 compares the embedding vector of each noun as a subject versus an object, to see how differently nouns behave in the two roles.

6 rgai.inf.u-szeged.hu/project/nlp/research/w2v/doc.html

cutoff	shape with unfilled	shape without unfilled
1	(324 196, 90 606, 287 967)	(206 488, 41 075, 188 619)
10	(160 629, 37 427, 129 694)	(109 432, 19 824, 92 635)
100	(92 999, 20 937, 69 536)	(71 768, 13 907, 57 420)
1000	(44 168, 10 444, 32 359)	(40 309, 8 838, 30 280)
10000	(13 765, 5 070, 12 313)	(13 610, 4 895, 12 115)
100000	(3 474, 2 313, 4 120)	(3 463, 2 308, 4 108)
1000000	(546, 814, 981)	(545, 813, 980)
10000000	(36, 194, 87)	(35, 194, 86)

Table 44: The length of each axis, i.e. the number of subjects, verbs, and objects, at different frequency cutoffs.

7.5.1 *Experimental setting: the corpus and the similarity task*

In our experiments, we took the occurrence counts of ⟨subject, verb[7], direct object⟩ triples from the automatically dependency-parsed (Nivre et al. 2016) English corpus DepCC (Panchenko et al. 2018), irrespectively of whether there were other arguments or adjuncts. Regarding empty fillers, we investigated two alternatives: including them (represented by a fixed string) or discarding them from our statistics. `tensorly` (Kossaifi et al. 2016) was used for CPD and (general and non-negative) Tucker decomposition of tensors. For tensor population in `COO`rdinate format, we use the `sparse` Python library.

Our quantitative tests are based on a classical similarity data-set for English transitive verb structures (SVO triples) by Kartsaklis and Sadrzadeh (2014, KS14). The data-set contains triples with gold (human) similarity scores. We represent SVO triples by concatenating the corresponding subject, verb, and object embedding vector (we experimented with normalizing the vectors, but we did not find it useful), and computed the Spearman correlation between the cosine similarities of the (long) vectors in each pair with the human scores.

7.5.2 *Quantitative results in transitive structure similarity*

We populated tensors with the association measures introduced in Section 7.2. The statistics were based on either including empty argument fillers (i.e. treating all arguments "optional") or excluding these occurrences. We took different cutoffs and computed non-negative or general CPD or Tucker decompositions in different ranks. Table 44 shows the length of each axis, i.e. the number of subjects, verbs, and objects, at different frequency cutoffs.

7 *Verb* means, in Universal Dependencies terms, that the `upos` starts with `VB`.

assoc measure	unfilled	cutoff	non-negative	decomp algo	rank	corr
pmi-sali	included	1 000 000	non-neg	parafac	64	0.7359
pmi-sali	included	1 000 000	non-neg	parafac	128	0.7097
pmi	included	1 000 000	non-neg	parafac	64	0.6857
pmi-sali	included	1 000 000	non-neg	parafac	32	0.6773
pmi-sali	included	*300 000*	non-neg	parafac	64	0.6630
npmi	included	1 000 000	non-neg	parafac	64	0.6602
dice-sali	included	1 000 000	non-neg	parafac	64	0.4709
pmi-sali	*excluded*	1 000 000	non-neg	parafac	64	0.4578
pmi-sali	included	1 000 000	*general*	parafac	64	0.4560
ldice	included	1 000 000	non-neg	parafac	64	0.4409
log-freq	included	1 000 000	non-neg	parafac	64	0.4322
iact-sali	included	1 000 000	non-neg	parafac	64	0.4112
niact	included	1 000 000	non-neg	parafac	64	0.4068
pmi-sali	included	*3 000 000*	non-neg	parafac	64	0.3936
iact	included	1 000 000	non-neg	parafac	64	0.3248
pmi-sali	included	1 000 000	non-neg	*tucker*	64	0.2989

Table 45: Quantitative results: correlations in the subject-verb-object triple similarity task (Kartsaklis and Sadrzadeh 2014) obtained with word embeddings of tensor decompositions.

Correlations we obtain in the subject-verb-object task are shown in Table 45. The properties of the original sparse tensor (the association measure, whether empty fillers are included, and the frequency cutoff) are show on the left of the vertical line, while those of the decompositions (non-negative or general CPD or Tucker decompositions to the specified rank) are shown on the right. The table shows, in addition to the the best setting, each setting obtained by changing one meta-parameter. The best result is obtained by non-negative CPD. The horizontal lines shows where our best general Tucker, general CPD, and non-negative Tucker decompositions – which we discuss later in this subsection, and are not shown in this table – end up. In Tucker decompositions, we use the same rank among all axes.

We obtained the best correlation, 0.7359, from the decomposition of a tensor populated with salience-weighted PMI values, including empty fillers, and setting the frequency cutoff to 1 million, i.e. restricting the axes of the tensor to the subjects, verbs, and objects that appear at least 1 million times. This best correlation was obtained with non-negative CPD in rank 64. This correlation value is in the same range as 0.76 obtained by Hashimoto et al. (2014) with a much more complex system that used to be the state-of-the-art, when this task was fashionable.

The table shows the correlation obtained by changing each (meta)-parameter. While the results seem to be relatively robust with respect to the decompositions *rank*, it may be interesting that when we concatenate the subject, the verb, and the object embedding vectors, 64

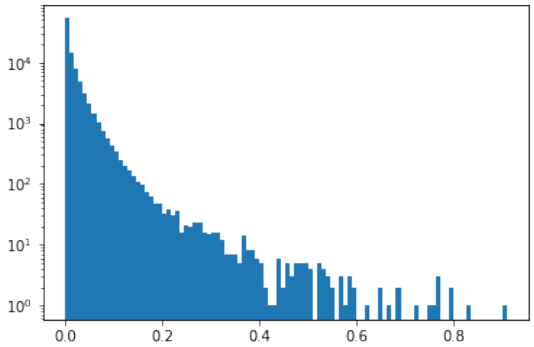

Figure 19: The histogram of the verb embedding matrix elements. Note that the vertical axis, which corresponds to the histogram count in each bin, is logarithmic. The figure suggests that frequency decreases faster than exponentially as larger weights are considered.

dimensional each, we get a vector in the famous range of a couple of hundreds of dimensions, which proved to work well in many different scenarios like LSA and static word embeddings (see the introduction).

As for our *association measures,* different weighted variants (salience, vanilla, or normalization) of PMI work the best, followed by log-Dice and log frequency. Variants of interaction information performs the worst.

The inclusion of empty fillers, the frequency cutoff, and the decomposition rank are all related to the *size of the tensors*. While we have already seen that the decomposition rank does not have a great influence on the results, if we exclude empty fillers, a more generous frequency cutoff may theoretically lead to better results than if we change only one of these two parameters. It turns out, that we can indeed get relatively good result (0.694181) this way, but with general Tucker decomposition (instead of non-negative CPD) and log-Dice (instead of salience-weighted). The cutoff is 1 million.

Non-negative decomposition is advantageous from the interpretational point of view, because in our experiments, they resulted in embedding matrices which are *sparse* in the broad sense that most coordinates are low. Figure 19 shows a histogram of the matrix elements. Note that the vertical axis, which corresponds to the histogram count in each bin, is logarithmic. The figure suggests that frequency decreases faster than exponentially as larger weights are considered. The good performance of non-negative CPD suggests that non-negativity introduces meaningful structure. Sparsity raises the hope that coordinate are interpretable, i.e. they correspond to concepts or properties.

CPD has the advantage that it maps the *modes in the same space*. In our case, this is the most interesting for subjects and objects: we can compare the same noun in the two roles. We return to this in Section 7.5.4.

While our best results have been obtained with non-negative CPD, we discuss general Tucker and CPD and non-negative Tucker as well. Results with general decompositions and non-negative Tucker are shown in Table 46 and Table 47, respectively. General Tucker and CPD and non-negative Tucker all prefer normalized PMI as the association measure, disfavor interaction information, and results with log frequency and log Dice vary. General and non-negative Tucker obtains the best results with the same rank as non-negative CPD, and the two non-negative decomposition algorithms also share the value for a best cut-off. It is inconclusive whether it is advantageous to include occurrences with unfilled arguments in our statistics.

7.5.3 *Qualitative analysis of latent dimensions*

Now we investigate the latent dimensions obtained by tensor decomposition. We experimented with non-negative and general CPD and Tucker decomposition with the hyper-parameters that reached the best result in the SVO-similarity task.

The latent dimensions are shown in Tables 48 to 50. (Dimensions with general Tucker are degenerate, and they omitted to save space.) Each line corresponds to a latent dimension. Dimensions are visualized by the words with the greatest coordinates in the dimension. Blocks represent dimension triples. ∅ denotes that the corresponding grammatical function is unfilled. Some latent dimensions, like the first one in our non-negative CPD are dominated by (the empty filler and) pronouns. In these cases we *emphasize* the first contentful filler. `-rrb-` stands for right round brackets, and its appearance may be an artifact of the corpus.

In the case of CPD, the dimensions are enumerated in the order as returned by the algorithm. With Tucker, the values g_{ijk} in the core tensor $\mathcal{G}$ represent the interaction between the ith latent dimension for subjects, the jth one for verbs, and the kth one for objects. We sorted the triples of SVO latent dimensions in our best non-negative and general Tucker decomposition by this interaction strength. The index of each dimension, as returned by the algorithm, is also shown in the table. E.g. the first block in non-negative Tucker shows that the strongest interaction is between the 5th latent dimension of subjects, the 10th one for verbs, and the 7th one for objects. Note that in the non-negative case, $g_{ijk} \geqslant 0$, so we do not have to take the absolute value. Dimensions obtained with the two *non-negative algorithms* seem semantically interpretable, while those from general decomposition are less convincing.

assoc measure	unfilled	cutoff	rank	correlation
npmi	included	100 000	64	0.7191
pmi-sali	included	100 000	64	0.7049
log-freq	included	100 000	64	0.6883
pmi	included	100 000	64	0.6759
npmi	included	30 000	64	0.6729
ldice	included	100 000	64	0.6685
ldice-sali	included	100 000	64	0.6666
npmi	included	300 000	64	0.6598
npmi	included	100 000	128	0.6540
npmi	included	100 000	32	0.6042
npmi	excluded	100 000	64	0.5207
iact-sali	included	100 000	64	0.5059
niact	included	100 000	64	0.4632
iact	included	100 000	64	0.4316

assoc measure	unfilled	cutoff	rank	correlation
npmi	excluded	300 000	256	0.6383
pmi-sali	excluded	300 000	256	0.6166
pmi	excluded	300 000	256	0.5811
npmi	excluded	1 000 000	256	0.5754
npmi	excluded	100 000	256	0.5713
npmi	excluded	300 000	512	0.5677
npmi	excluded	300 000	128	0.5290
npmi	excluded	30 000	256	0.5239
npmi	included	300 000	256	0.5070
log-freq	excluded	300 000	256	0.2465
ldice	excluded	300 000	256	0.2093
iact-sali	excluded	300 000	256	0.1280
niact	excluded	300 000	256	0.0726
iact	excluded	300 000	256	0.0615

Table 46: Results with general Tucker (top) and general CPD (bottom).

assoc measure	unfilled	cutoff	rank	correlation
npmi	excluded	1 000 000	64	0.5186
npmi	excluded	1 000 000	128	0.5102
npmi	excluded	300 000	64	0.4814
pmi	excluded	1 000 000	64	0.4563
pmi-sali	excluded	1 000 000	64	0.4387
npmi	excluded	1 000 000	32	0.3753
npmi	excluded	3000 000	64	0.3366
npmi	optional	1 000 000	64	0.2889
iact	excluded	1 000 000	64	0.0989
log-freq	excluded	1 000 000	64	0.0763
ldice	excluded	1 000 000	64	0.0698
ldice-sali	excluded	1 000 000	64	0.0619
niact	excluded	1 000 000	64	0.0454
iact-sali	excluded	1 000 000	64	0.0064

Table 47: Results with non-negative Tucker.

dim	words
0	∅, that, which, it, *story*, he, they, who, what, one, she, work, event, -rrb-, this, you...
0	catch, attract, draw, pay, deserve, capture, gain, grab, get, receive, focus, require,...
0	attention, eye, crowd, interest, fire, visitor, audience, conclusion, breath, people, ...
1	∅, who, we, he, I, you, she, they, -rrb-, *student*, member, people, group, Center, parti...
1	attend, host, hold, organize, schedule, enjoy, join, arrange, cancel, miss, watch, pla...
1	meeting, event, conference, session, party, show, school, class, dinner, church, tour,...
2	that, which, it, this, ∅, *change*, factor, they, choice, condition, decision, issue, -rr...
2	affect, impact, influence, improve, hurt, reflect, benefit, change, damage, enhance, a...
2	ability, performance, health, outcome, life, quality, result, business, development, e...
3	file, which, page, site, that, it, book, report, section, document, collection, websit...
3	contain, include, provide, have, list, feature, display, show, comprise, present, give...
3	information, link, material, number, list, datum, name, content, statement, reference,...

Table 48: Latent dimensions with Non-negative ParaFac

dim	words
5	court, Court, judge, panel, official, we, he, it, authority, government, -rrb-, Board,...
10	reject, dismiss, deny, grant, hear, consider, decide, accept, throw, resolve, sustain,...
7	motion, appeal, claim, request, argument, case, challenge, application, complaint, att...
4	revenue, sale, share, price, stock, production, cost, rate, order, volume, number, fut...
3	rise, fall, increase, jump, drop, decline, climb, decrease, grow, gain, slip, represen...
1	percent, %, $, increase, point, most, rate, level, average, less, matter, value, cost,...
11	hotel, property, room, restaurant, home, Center, house, location, facility, House, are...
8	offer, boast, feature, have, provide, include, enjoy, serve, accommodate, occupy, prep...
9	room, pool, accommodation, access, facility, restaurant, variety, service, view, range...
6	board, Council, Board, Commission, Committee, member, committee, Congress, Court, cour...
2	approve, adopt, reject, pass, consider, review, endorse, propose, award, recommend, ac...
2	resolution, request, budget, plan, proposal, contract, change, application, project, i...

Table 49: Latent dimensions with Non-negative Tucker

dim	words
0	Israel, group, government, Foundation, Association, company, -rrb-, military, army, Cl...
0	launch, wage, suspend, mount, begin, run, fund, organize, sponsor, administer, carry, ...
0	campaign, attack, program, initiative, operation, strike, programme, website, effort, ...
1	user, you, application, customer, developer, visitor, client, processor, device, User,...
1	access, select, specify, upload, view, enter, edit, browse, click, create, retrieve, m...
1	file, datum, content, document, page, parameter, site, folder, node, Internet, informa...
2	device, assembly, means, structure, system, element, plate, section, interface, unit, ...
2	comprise, include, contain, have, utilize, employ, represent, say, mean, control, enab...
2	layer, element, device, tube, housing, spring, electrode, pump, plate, container, memb...
3	attorney, plaintiff, defendant, party, respondent, prosecutor, State, lawyer, governme...
3	file, receive, oppose, make, give, present, withdraw, handle, publish, drop, provide, ...
3	motion, notice, petition, appeal, response, answer, objection, charge, request, submis...

Table 50: Latent dimensions with General ParaFac

7.5.4 *Comparing subject and object vectors*

Tensor decomposition can shed light on how differently nouns behave as subjects and as objects. This question is related to symmetric factorization (Bailey, Meyer, and Aeron 2018), which imposes symmetry constraints between the embeddings of the same entities in different modes (in our case, between the embeddings of the same noun as a subject or an object). Our approach is complementary, based on that CPD maps nouns as subjects and objects in the same space.

In our experiments, we consider (non-negative) CPD decomposition with the hyper-parameters that proved best in English SVO-similarity. We computed the (unnormalized) dot product similarity between the subject and object vector of each noun, and sorted all the nouns by this similarity. The largest distance is found with *∅, he, she, they, I, device, system, that, you, it...*, while the most symmetric nouns are *doubt, reality, future, same, hope, feeling, mine, reason, consumer, plenty...* A possible explanation is that the former lemmas, especially personal pronouns (or their inflected forms), are much more frequent in agentive roles than other nouns, while they are infreqent in patient roles. Words in the second group can be framed in language both as animate and as inanimate. *Future* or *hope* are not alive in the biological sense, but they are often attributed agentive roles (what can be called a metaphorical use of language but being metaphorical dos not mean that the usage is peripheral, as it has been noted by linguists).

7.6 CONCLUSION OF THE MAIN EXPERIMENTS

Weighted variants of positive pointwise mutual information proved better than the considered alternatives in modeling subject-verb-object structure similarity. It does not matter whether we include occurrences with unfilled arguments in our statistics. Our best results were obtained with non-negative CPD. The best frequency cutoff and the decomposition rank is the same for the two non-negative decomposition algorithms, which raises the hope that these hyper-parameters of non-negative CPD can be fine-tuned based on the much faster non-negative Tucker, but this needs to be tested in other setups. Our experiments provided lexically interpretable latent dimensions, and the difference between subject and object embeddings can be related to animacy, at least in the case of non-negative CPD.

7.7 FOLLOW-UP

In this section, we report experiments, which did not appear in Makrai (2022).

# verbs	verbs
702	have, do, get, go, take, think, know, want, need, give, look, work, provide, try, ...
131	live, talk, stand, die, walk, wait, sit, stay, wonder, care, arrive, fly, gon, sleep, ...
86	kill, catch, trust, bear, email, marry, fuck, date, judge, bless, honor, forgive, beg,...
85	add, eat, produce, deliver, prepare, drink, spread, cook, burn, taste, wash, supply, ...
80	use, develop, manage, perform, complete, replace, install, connect, test, conduct, ...
80	let, reach, hit, cost, exceed, rate, approach, /, -lsb-_VBD, rank, -lsb-_VB, \, -lsb-_...
79	put, break, pull, throw, push, lay, stick, grab, touch, press, suck, kick, shake, ...
77	identify, commit, defend, repeat, expose, separate, dig, heal, dress, distinguish, ...
76	send, check, view, click, display, generate, update, access, search, store, delete, ...
65	leave, enter, visit, fill, explore, ride, clean, cross, surround, locate, clear, rent,...
59	be, come, start, happen, seem, begin, continue, appear, lead, end, occur, prove, ...
58	help, keep, bring, remind, hurt, strike, worry, blow, inspire, bother, surprise, suit,...
57	tell, ask, call, thank, please, join, contact, become, assist, hire, name, engage, ...
51	pay, spend, save, raise, determine, compare, charge, measure, adjust, predict, invest,...
46	make, see, find, love, like, hear, enjoy, remember, miss, guess, recommend, notice, ...
43	understand, discover, recognize, examine, evaluate, investigate, acknowledge, assess, ...
43	face, experience, address, fix, handle, suffer, solve, celebrate, resolve, mark, ...
39	receive, win, lose, earn, gain, extend, deserve, capture, retain, lack, exercise, ...
37	plan, fail, focus, vote, act, deal, attempt, rely, struggle, participate, benefit, ...

Table 51: Verb clusters obtained from our verb embedding vectors in an unsupervised fashion. The smallest cluster is omitted to save space.

7.7.1 *Clustering verb vectors*

Semantic *classes* of verbs like those in VerbNet (Kipper et al. (2008), which are refinements of Levin (1993)'s classes) may be induced by clustering verb embedding vectors. If clusters obtained in unsupervised fashion correspond to gold verb classes, ambiguous verbs like *play* mentioned in Section 7.1 may be detected as outliers from the clusters, as their uses are composed of occurrences corresponding to different clusters.

Our method for obtaining verb clusters consists of mapping verb embedding vectors to a lower dimensional space with UMAP (McInnes et al. 2018) and clustering them with HDBScan (McInnes, Healy, and Astels 2017), which is a hierarchical, density based clustering algorithm. Dimensionality reduction is needed because density makes little sense in hundreds of dimensions. Our choices of UMAP meta parameters are the following: We map verb embedding vectors to 16 or 32 dimensions (fine-tuned in a comparison to VerbNet, see later). In HDBScan, we set the number of neighbors to 30 and the minimum distance to 0, following the recommendations at `readthedocs`[8]. The metric in the ambient space (i.e. the original, high-dimensional one) is cosine. Minimum cluster size is 15 or 5, and the related parameter of `min_samples` is 5.

We compare non-negative and general CPD and Tucker decompositions. The parameters of the tensor and its decompositions are set to

8 https://umap-learn.readthedocs.io/en/latest/clustering.html#umap-enhanced-clustering

preverb	verb	args					gloss
∅	bíz(ik)	NOM		-bAn 'in'			trust sth
(rá) 'onto'	bíz	NOM	ACC		-rA 'onto'		entrust sg to sy
meg PERFECT	bíz(ik)	NOM		-bAn 'in'			trust sy
meg PERFECT	bíz	NOM	ACC			INS	entrust sy with sg
el 'away'	bíz(za)	NOM	self-ACC				get conceited

Table 52: Argument structure variants of the Hungarian verb *bíz(ik)* based on Szécsényi (2019).

the value with the best score in the SVO-similarity task. We set one hyper-parameters of UMAP and HDBScan each, namely the dimension we map to and minimum cluster size, based on comparison to VerbNet classes. In these computation we take VerbNet from the `nltk.corpus` package. In many cases, there are more class IDs associated to a verb. We take the first one, as returned by the corresponding function. Out-of-vocabulary verbs are treated as a separate class. We compare are clustering to VerbNet classes with adjusted rand score in scikit-learn (Pedregosa et al. 2011). We get the greatest score with non-negative Tucker (embeddings mapped to 16 dimensions, and minimum cluster size set to 15).

Table 51 shows the greatest clusters of English verbs. The greatest cluster, separated by a line in the table is the one called `-1` in HDBScan. It contains points that "fall out" (as members of very small would-be clusters) in the hierarchy. The algorithm considers them outliers[9]. In our case, it seems that they are general verbs, especially those that we find in light verb constructions. The remaining clusters seem to be semantically coherent.

7.7.2 *Hungarian data and preverbs*

We propose hypotheses for future work. In Hungarian, there are two phenomena that interfere with verb agument structure and ambiguity. Table 52, based on Szécsényi (2019), illustrates these with the verb *bíz(ik)* 'trust'. We can see that preverbs (verb particles, which can modify both the aspect and the meaning of a verb (Ágnes Kalivoda 2021)) interfere with verb meaning, and the apparently incidental appearance of the suffix *-ik* (which can be argued to be related to unaccusativity) increases data sparsity. In our preliminary experiments, we built a *subject* × *preverb* × *verb* × *object* tensor from verb constructions in the data-base of the Mazsola verb argument browser (Bálint Sass 2015). In this earlier, unpublished phase of the project, we used CPD decomposition, solved by the Orth-ALS (Sharan and Valiant 2017) algorithm. For the future, we suggest introducing a mode for *-ik*. The "vocabulary" of this axis would consists of only two choices: with or without

9 See https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html#

-ik. The hypothesis is that this tensor would profit from denser data representation.

7.8 CONCLUSION

Tensor decompositions offers a direction orthogonal to the mainstream (Rogers, Kovaleva, and Rumshisky 2020) in the data-driven understanding of linguistic structure. We may want to learn semantic verb classes in an unsupervised fashion. If verb embedding vectors correspond to Levin's (1993) verb classes, ambiguous verbs could be identified in the form of outliers in the clustering. This line of research can be extended cross-lingually (Vulić, Mrkšić, and Korhonen 2017; Majewska et al. 2018; Sun et al. 2010).

The name of the song is called "Haddocks' Eyes."'
'Oh, that's the name of the song, is it?' Alice said, trying to feel interested.
'No, you don't understand,' the Knight said, looking a little vexed. 'That's what the name is called. The name really is "The Aged Aged Man."'
'Then I ought to have said "That's what the song is called"?' Alice corrected herself.
'No, you oughtn't: that's quite another thing! The song is called "Ways and Means": but that's only what it's called, you know!'
'Well, what is the song, then?' said Alice, who was by this time completely bewildered.
'I was coming to that,' the Knight said. 'The song really is "A-sitting On A Gate": and the tune's my own invention.'

— Lewis Carroll

8 CROSS-LINGUAL WORD SENSE INDUCTION

Contents

This final chapter investigates the connection between word ambiguity and multilinguality/translation. Section 8.1, which originally appeared as Makrai (2016b), attempts lexical induction with triangulation, a multilingual method which is sensitive to word ambiguity. The remainder of the chapter, which originally appeared as Borbély, Kornai, Makrai, and Nemeskey (2016) and Makrai and Lipp (2018), offers an evaluation method for multisense (static) word embeddings, which

have represented different meanings of words with different vectors before deep language models (a.k.a. contextualized word representations, Section 4.3).

8.1 FILTERING WIKTIONARY TRIANGLES

Triangulation infers word translations in a pair of languages based on translations to other, typically better resourced ones called pivots. This method may introduce noise if words in the pivot are polysemous. The reliability of each triangulated translation has traditionally been estimated by the number of pivot languages (Tanaka and Umemura 1994).

Mikolov, Le, and Sutskever (2013) introduce a method for scoring word translations. Translation is formalized as a linear mapping between distributed vector space models (VSM) of the two languages. VSMs are trained on monolingual data, while the mapping is learned in supervised fashion, using a seed dictionary of some thousand word pairs.

In this section, we apply linear mapping to filter triangulated translations, and show that scores by the mapping are smoother measure of merit than the number of pivots. The methods we use are language-independent, and the training data is easy to obtain for many languages. For research reported in this section, we chose the German-Hungarian pair for evaluation, in which the filtered triangles resulting from our experiments were, to the best of our knowledge, the greatest freely available list of word translations by the time.

8.1.1 *Introduction*

Word translations arise in dictionary-like organization as well as via machine learning from corpora. The former is exemplified by Wiktionary, a crowd-sourced dictionary with editions in many languages. Ács et al. (2013) obtain word translations from Wiktionary with the pivot-based method, also called triangulation, that infers word translations in a pair of languages based on translations to other, typically better resourced ones called pivots. Triangulation may introduce noise if words in the pivot are polysemous. The reliability of each triangulated translation is basically estimated by the number of pivot languages (Tanaka and Umemura 1994).

Mikolov, Le, and Sutskever (2013) introduce a method for generating or scoring word translations. Translation is formalized as a linear mapping between distributed vector space models (VSM) of the two languages. VSMs are trained on monolingual data, while the mapping is learned in a supervised fashion, using a seed dictionary of some thousand word pairs. The mapping can be used to associate existing translations with a real-valued similarity score.

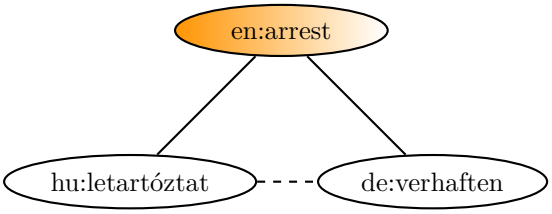

Figure 20: Triangulation

The project reported in this section exploits human labor in Wiktionary combined with distributional information in VSMs. We train VSMs on gigaword corpora, and the linear translation mapping on direct (non-triangulated) Wiktionary pairs. This mapping is used to filter triangulated translations based on scores. The motivation is that scores by the mapping may be a smoother measure of merit than considering only the number of pivot for the triangle. We evaluate the scores against dictionaries extracted from parallel corpora (Tiedemann 2012). We show that linear translation really provides a more reliable method for triangle scoring than pivot count.

The methods we use are language-independent, and the training data is easy to obtain for many languages. We chose the German-Hungarian pair for evaluation, in which the filtered triangles resulting from our experiments are the greatest freely available list of word translations we are aware of.

8.1.2 *Triangulation*

A method for creating dictionaries is triangulation through better-resourced ones called the *pivot* (Tanaka and Umemura 1994). The idea is that if the English translation of the Hungarian word *letartóztat* is *arrest*, and the German translation of *arrest* is *verhaften*, then the German translation of *letartóztat* is *verhaften*, see Figure 20.

Triangles are corrupted by ambiguity in the pivot word (the one in the middle): German *Dose* can be translated as *can* to English (as a synonym of *tin*), which, as a verb, translates to *tud* in Hungarian, which is unrelated to *Dose*. Saralegi, Manterola, and Vicente (2011) analyze two methods for pruning wrong triangles: one based on exploiting the structure of the source dictionaries, and the other based on distributional similarity computed from comparable corpora. The project reported in this section is more similar to the later in that it uses distributional information applying a method connected to neural language modeling.

8.1.3 *Linear translation*

As we already mentioned in Section 4.2, Mikolov, Le, and Sutskever (2013) discovered that VSMs of different languages have such similari-

ties that a linear mapping can map representations of source language words to the representation of their translations. The method belongs to the paradigm of supervised machine learning: specifically it makes use of a great amount of monolingual data i.e. gigaword corpora for training, needing to be supervised by a seed dictionary of some thousand words. Mikolov et al. formalize translation as linear mapping $W \in \mathbb{R}^{d_2 \times d_1}$ from the source (monolingual) VSM $\mathbb{R}^{d_1}$ to the target one $\mathbb{R}^{d_2}$: the translation $z_i \in \mathbb{R}^{d_2}$ of a source word $x_i \in \mathbb{R}^{d_1}$ is approximately its image Wx_i by the mapping. The translation model is trained with linear regression on the seed dictionary

$$\min_W \sum_i ||Wx_i - z_i||^2$$

and can be used to collect translations for the whole vocabulary (by choosing z_i to be the nearest neighbor of Wx_i) or to score a translation z coming from some other source (with the score being the distance between Wx_i and z_i).[1] In the original setting of the collection mode, evaluation is done on another thousand seed pairs.

A common error in linear translation is when there are target words that are returned as the translation of many words, which is wrong in most of the cases. Dinu, Lazaridou, and Baroni (2015) propose a method for downplaying the importance of such target words they call *global correction*. Our experiments use this method.

8.1.4 *Data*

Direct and triangulated Wiktionary translations have been extracted with wikt2dict[2] (Ács, Pajkossy, and Kornai 2013) that handles 43 editions of Wiktionary.

The German VSMs have been trained on SdeWaC (Baroni et al. 2009) and the Hungarian on the concatenation of the Hungarian Webcorpus (Halácsy et al. 2004) and the Hungarian National Corpus (Oravecz, Váradi, and Sass 2014) with `word2vec`[3] (Mikolov, Chen, et al. 2013).[4]

For training and using the linear mapping, we forked[5] the implementation by Dinu, Lazaridou, and Baroni (2015). The German to

1 Mikolov et al. use a surprising combination of vector distances, Euclidean distance in training and cosine similarity (and distance) in collection (and, respectively, scoring) of translations. This choice is theoretically unmotivated, but we (Makrai 2015) also found it to work better than more consistent combinations of metrics. However, see Xing et al. (2015) for opposing results.

2 https://github.com/juditacs/wikt2dict

3 https://code.google.com/p/word2vec/

4 The German VSM has been a continuous bag of words model in 300 dimensions (infrequent words have been cut off at 100 occurrences), the Hungarian a 600 dimensional one (with a cut-off of 10). The choice of meta-parameters was not fully systematic.

5 https://github.com/makrai/dinu15/

documents	3208
sentences	3.2 M
German tokens	23.3 M
Hungarian tokens	19.7 M
extracted word pairs	29.1 K

Table 53: The German Hungarian subsection of the OpenSubtitles2013 parallel corpus (Tiedemann 2012)

Hungarian mapping was trained on the 5K direct word pairs that are supported by the most pivots in Wiktionary. All the triangles were scored. The Hungarian word embedding (and some glue code we wrote for this project) is freely available[6].

The scoring has been evaluated against a dictionary in the OPUS project[7] that has been extracted by Tiedemann (2012) from the OpenSubtitles2013 parallel corpus, a collection of translated movie subtitles[8]. OpenSubtitles2013 contains 59 languages. The sizes of the German Hungarian subsection are shown in Table 53.

Most of our training data are general in their *domain*: web corpora (SdeWaC, the Hungarian Webcorpus), a curated corpus (the Hungarian National Corpus, as far as a corpus of 754 million words may be curated), and a crowd-sourced but otherwise standard dictionary (Wiktionary). One may ask whether the domain of the reference dictionary extracted from movie subtitles is general to an appropriate extent, or how far a problem of domain mismatch between train and test may arise. We hypothesize that the mismatch is negligible and defer a more detailed analysis to further research.

8.1.5 *Evaluation*

We evaluated the vector-based scoring of triangulated translational word pairs (*triangles*) in comparison with a dictionary created from the parallel corpus OpenSubtitles2013. For each (German) word, we consider as gold translations all the (Hungarian) words that are listed in the OpenSubtitles2013 dictionary as its translation.

For evaluation, we sort the triangles in two orders: as baseline, by the number of pivots for the triangle, and more importantly, by the score in the linear mapping (cos). Then in each order, we compute accuracy on each 1000-word slice of the list (e.g. triangles 1–1000, then 1001–2000, etc.) taking OpenSubtitles2013 translations as gold.

While overall accuracy of the linear scoring (8.58%) is slightly worse than that of pivot counting (9.32%), Figure 21 suggests that in sort

6 https://github.com/makrai/efnilex-vect
7 http://opus.lingfil.uu.se/
8 http://www.opensubtitles.org/

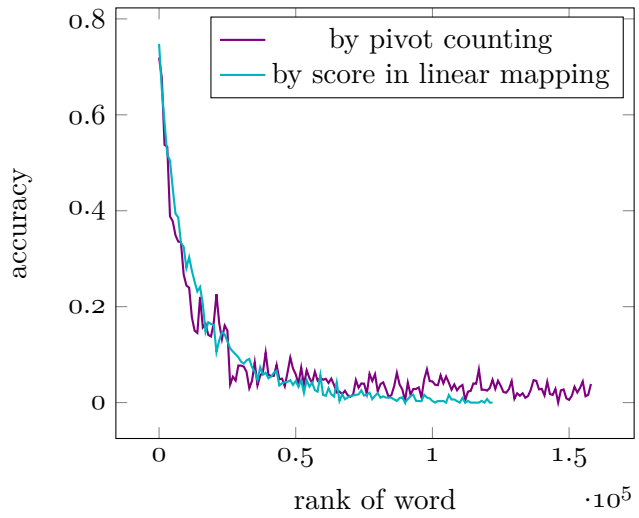

Figure 21: Accuracy curve of triangles sorted by their pivot count as baseline, or score in linear translations (cos). The later is smoother.

by cos, accuracy descends more smoothly than in sort by pivot count. (The last 22.73% of the nearly 160 K triangles is out of the vocabulary of one or both of the VSMs, so cos cannot be computed.) Now we turn to a more quantitative support of this visual analysis.

8.1.5.1 *Quantitative analysis of smoothness*

We measure the smoothness of the accuracy curves by how well they can be approximated by a function in some parametric family, see Figures 22 to 25. We tried two families with similar results. The fist family is exponential functions of the form

$$a \cdot \exp(-bx) + c,$$

where x is the index of the vocabulary slice (0 for words 0–1000, 1 for 1001–2000, etc), and a, b, and c are parameters to fit. The second family is that of power law functions

$$a \cdot (bx + c)^k,$$

where k is another parameter to fit, and the remaining variables play similar roles as in the exponential case. The error of the fit (i. e. the lack of smoothness) is quantified as the mean squared error (MSE) between the two curves. The MSE of the two accuracy curves (scoring translations by pivot counting or cosine score) approximated by the two families (exponential or power law functions) are shown in Table 54. The MSE of the accuracy curve in pivot counting is 2.51 (resp. 4.42) times more than that in scoring by the linear mapping, when both curves are modeled as exponential (resp. power law) functions. It is probably also worth mentioning that the accuracy is slightly better for 20–30 000 higher-ranked words in the proposed method than in the baseline, see Figures 26 and 27.

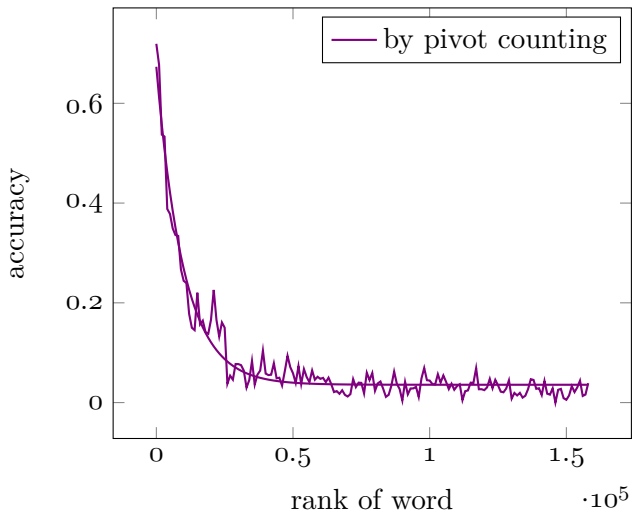

Figure 22: The accuracy curve of pivot counting approximated by an exponential function.

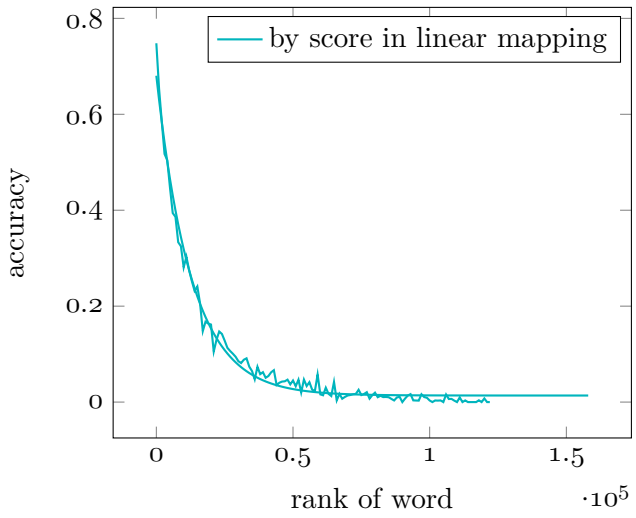

Figure 23: The accuracy curve of scores by the linear mapping approximated by an exponential function.

scoring method	exp	power law
pivot counting	6.1859e-04	5.2182e-04
linear mapping	**2.4574e-04**	**1.1789e-04**
ratio	2.51	4.42

Table 54: The mean squared error of fitting parametric curves to the accuracy values obtained by translation scoring methods. Linear mapping produces a smoother accuracy decay than pivot counting.

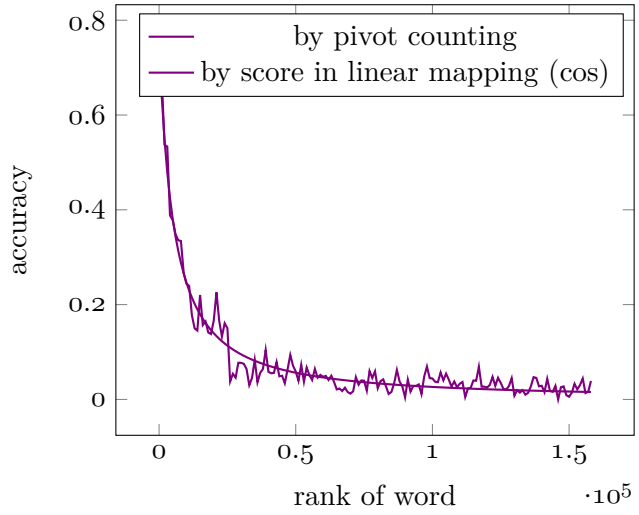

Figure 24: Accuracy curves of scores by pivot count approximated by power law functions.

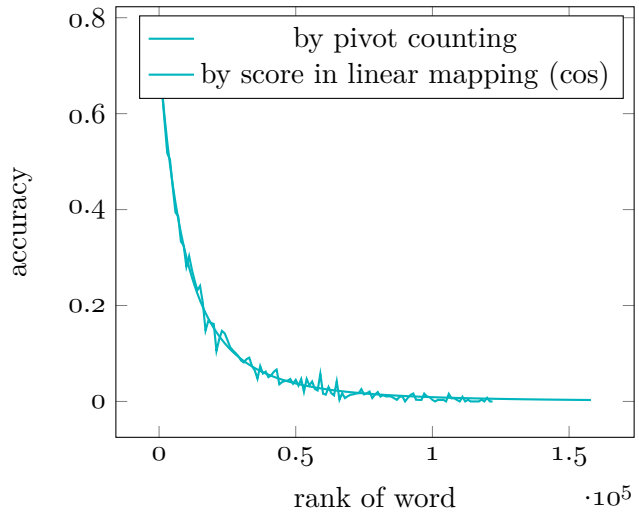

Figure 25: Accuracy curves of scores by the linear mapping approximated by power law functions.

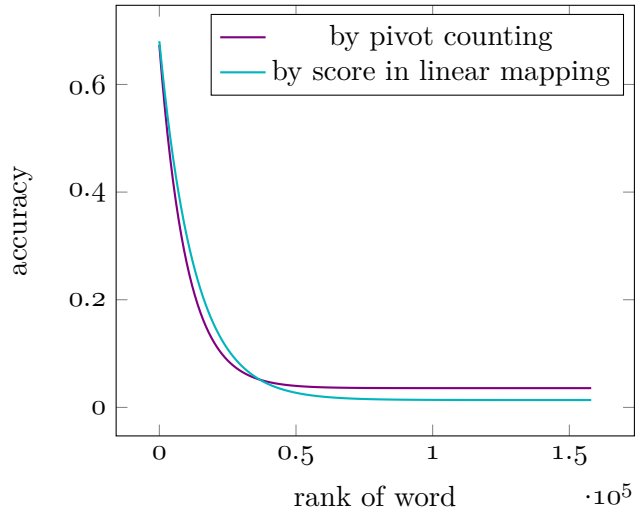

Figure 26: The exponential approximations of the accuracy curves.

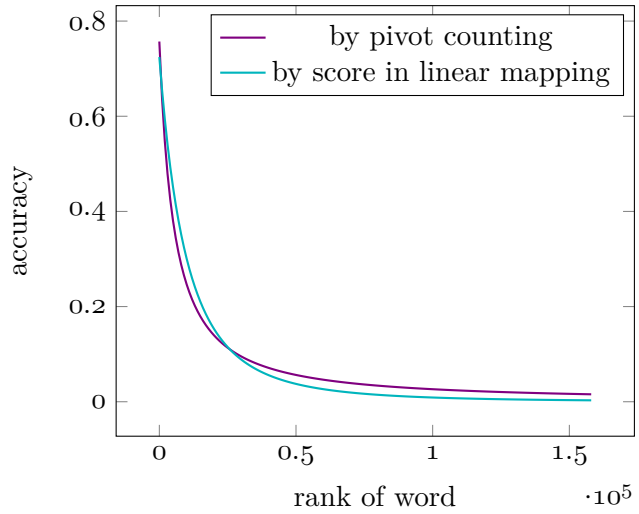

Figure 27: The power law approximations of the accuracy curves.

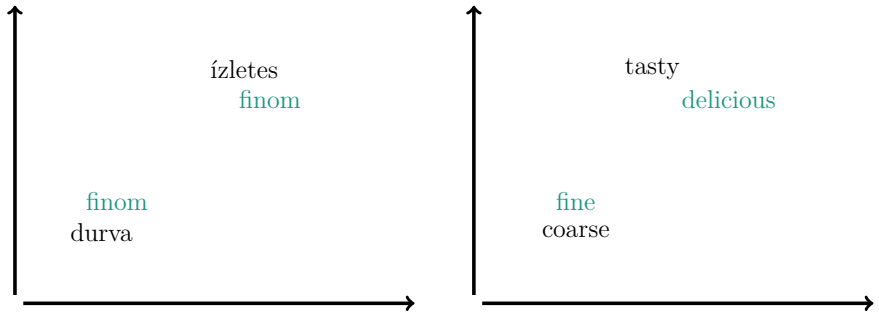

Figure 28: Linear translation of word senses. The Hungarian word *finom* is ambiguous between 'fine' and 'delicious'.

8.2 DO MULTI-SENSE EMBEDDINGS LEARN MORE SENSES?

Multi-sense word embeddings (MSEs) have modeled different meanings of word forms with different vectors since before the advent of deep language models (a.k.a. contextualized word representations, Section 4.3). In this final section on the thesis, which originally appeared as Borbély, Kornai, Makrai, and Nemeskey (2016) and Makrai and Lipp (2018), we propose a method for evaluating MSEs by their degree of *semantic resolution,* measuring the detail of the sense clustering. The method exploits the principle that words may be ambiguous as far as the postulated senses translate to different words in some other language. In the context of embedding-based dictionary induction, we also test whether the orthogonality constraint and related vector preprocessing techniques help in reverse nearest neighbor search. These questions receive a negative answer.

8.3 TOWARDS A LESS *delicious* INVENTORY

Word sense induction (WSI) is the task of discovering senses of words without supervision (Schütze 1998). The goal of WSI can be set at two levels. We may more modestly aim to distinguish homophony from polysemy. Ideally, we could even differentiate between metonymy and metaphor, two subtypes of polysemy, discussed in more detail by Veronika Lipp in the next section. Approaches include multi-sense word embeddings (MSEs), i.e. vector space models of word distribution with more vectors for ambiguous words. In MSEs, each vector is supposed to correspond to a different word sense, but in practice, models frequently have different sense vectors for the same word form without an interpretable difference in meaning.

Our first publication in this topic (Borbély, Makrai, et al. 2016) appeared at the 1st Workshop on Evaluating Vector-Space Representations for NLP. Gladkova and Drozd (2016) calls polysemy "the elephant in the room" as far as evaluating embeddings are concerned. We

attacked this problem head on, by proposing a method for evaluating multi-sense word embeddings (MSEs).

We emphasize at the outset that our evaluation proposal probes an aspect of MSEs, *semantic resolution*, which is not well measured by the well-known word sense disambiguation (WSD) task that aims at classifying occurrences of a word form to different elements of a sense inventory pre-defined by some experts. Our goal is to probe the granularity of the inventory itself.

As we discussed in Section 3.1.1, the central linguistic/semantic/psychological property we wish to capture is that of a *concept*, the underlying word sense unit. To the extent standard lexicographic practice offers a reasonably robust notion (this is of course debatable, but we consider a straight correlation of 0.27 and and a frequency-effect-removed correlation of 0.60 over a large vocabulary[9] a strong indication of consistency), this is something that MSEs should aim at capturing. We leave the matter of aligning word senses in different dictionaries for future work, but we expect that by (manual or automated) alignment the inter-dictionary (inter-annotator) agreement can be improved considerably, to provide a more robust gold standard.

The differentiation of word senses is fraught with difficulties, especially when we wish to distinguish homophony, using the same written or spoken form to express different concepts, such as Russian *mir* 'world' and *mir* 'peace' from polysemy, where speakers feel that the two senses are very strongly connected, such as in Hungarian *nap* 'day' and *nap* 'sun'. To quote Zgusta (1971) "Of course it is a pity that we have to rely on the subjective interpretations of the speakers, but we have hardly anything else on hand". Etymology makes clear that different languages make different lump/split decisions in the conceptual space, so much so that translational relatedness can, to a remarkable extent, be used to recover the universal clustering (Youn et al. 2016).

One of the confounding factors is part of speech (POS, recall Section 3.1.2). Very often, the entire distinction is lodged in the POS, as in *divorce* (noun) and *divorce* (verb), while at other times this is less clear, compare the verbal *to bank* 'rely on a financial institution' and *to bank* 'tilt'. Clearly the former is strongly related to the nominal *bank* 'financial institution' while the semantic relation 'sloping sideways' that connects the tilting of the airplane to the side of the river is somewhat less direct, and not always perceived by the speakers. The Collins-COBUILD (CED, Sinclair (1987)) dictionary starts with the semantic distinctions and subordinates POS distinctions to these, while the Longman dictionary (LDOCE, Boguraev and Briscoe (1989)) starts with a POS-level split and puts the semantic split below. Of the Hungarian lexicographic sources, the Comprehensive Dictionary of Hungarian

9 These results are published in the same Borbély, Makrai, et al. (2016), but this thesis does not discuss them in detail, because they were conducted mainly by Dávid Nemeskey.

(NSZ, Ittzés (2011)) is closer to CED, while the Explanatory Dictionary of Hungarian (EKSZ, Pusztai (2003)), is closer to LDOCE in this regard.

Our method is based on the principle that words may be ambiguous to the extent to which their postulated senses translate to different words in some other language. For the translation of words, we applied the method by Mikolov, Le, and Sutskever (2013) who train a translation mapping from the source language embedding to the target as a least-squares regression supervised by a seed dictionary of the few thousand most frequent words. The translation of a source word vector is the nearest neighbor of its image by the mapping in the target space. In the multi-sense setting, we have translated from MSEs. (The target embedding remained single-sense.)

Section 8.3 discusses our linguistic motivation and section 8.4 introduces MSEs. In section 8.5, we elaborate on the cross-lingual evaluation. Part of the evaluation task is to decide on empirical grounds whether different good translations of a word are synonyms or translations in different senses. Reverse nearest neighbor search, the orthogonality constraint on the translation mapping, and related techniques are also discussed. Section 8.6 offers experimental results with quantitative and qualitative analysis. It should be noted that our evaluation is not very strict, but rather a process of looking for something conceptually meaningful in present-day unsupervised MSE models. We make our Hungarian multi-sense embeddings[10] and the code for these experiments[11] available on the web.

8.4 MULTI-SENSE WORD EMBEDDINGS

Vector-space language models with more vectors for each meaning of a word originate from Reisinger and Mooney (2010). Huang et al. (2012) trained the first neural-network-based MSE. Both works use a uniform number of clusters for all words that they select before training as potentially ambiguous. The first system with adaptive sense numbers and an effective open-source implementation is a modification of skip-gram (Mikolov, Sutskever, et al. 2013), *multi-sense* skip-gram by Neelakantan et al. (2014), where new senses are introduced during training by thresholding the similarity of the present context to earlier contexts.

Bartunov et al. (2016) and Li and Jurafsky (2015) improve upon the heuristic thresholding by formulating text generation as a Dirichlet process. In `AdaGram` (Bartunov et al. 2016), senses may be merged as well as allocated during training. `mutli`-sense skip-gram[12] (Li and Jurafsky 2015) applies the Chinese restaurant process formalization of

10 https://hlt.bme.hu/en/publ/makrai17

11 https://github.com/makrai/wsi-fest

12 Note the $l \leftrightarrow t$ metathesis in the name of the repo which is the only way of distinguishing it from the other two multi-sense skip-gram models.

the Dirichlet process. `neela`, `AdaGram`, and `mutli` have a parameter for semantics resolution (more or less senses): λ, α, and γ, respectively.

MSEs are still in the research phase: Li and Jurafsky (2015) demonstrate that, when meta-parameters are carefully controlled for, MSEs introduce a slight performance boost in semantics-related tasks (semantic similarity for words and sentences, semantic relation identification, part-of-speech tagging), but similar improvements can also be achieved by simply increasing the dimension of a single-sense embedding.

8.5 LINEAR TRANSLATION FROM MSES

As we already mentioned in Section 8.1.3, Mikolov, Le, and Sutskever (2013) discovered that embeddings of different languages are so similar that a linear transformation can map vectors of the source language words to the vectors of their translations.

The method uses a seed dictionary of a few thousand words to learn translation as a linear mapping $W \ \mathbb{R}^{d_1} \rightarrow \mathbb{R}^{d_2}$ from the source (monolingual) embedding to the target: the translation $z_i \in \mathbb{R}^{d_2}$ of a source word $x_i \in \mathbb{R}^{d_1}$ is approximately its image Wx_i by the mapping. The translation model is trained with linear regression on the seed dictionary

$$\min_W \sum_i ||Wx_i - z_i||^2$$

and can be used to collect translations for the whole vocabulary by choosing z_i to be the nearest neighbor (NN) of Wx_i. We follow Mikolov, Le, and Sutskever (2013) in (i) using different metrics, Euclidean distance in training and cosine similarity in collection of translations, and in (ii) training the source model with approximately three times greater dimension than that of the target embedding.

In a multi-sense embedding scenario, Borbély, Kornai, Makrai, and Nemeskey (2016)[13] take an MSE as the source model, and a single-sense embedding as target. The quality of the translation has been measured by training on the most frequent 5k word pairs and evaluating on another 1k seed pairs.

8.5.1 *Reverse nearest neighbor search*

A common problem when looking for nearest neighbors in high-dimensional spaces (Radovanović, Nanopoulos, and Ivanović 2010; Suzuki et al. 2013;

13 The 2016 paper measured the sense granularity with two methods: Section 2 was based on computer readable lexica, and Section 3 presented the multilingual method. The former was the work of Nemeskey. The latter is the joint work of Borbély and Makrai, with equal contribution. In the 2018 paper, Makrai elaborated the multilingual method alone.

		8192				16384				32768			
		general linear		orthogonal		general linear		orthogonal		general linear		orthogonal	
		any	disamb	any	disamb	any	disamb	any	disamb	any	disamb	any	disamb
fwd	vanilla	28.7%	2.40%	32.1%	2.40%	36.2%	3.40%	42.0%	4.70%	36.7%	4.20%	44.5%	6.00%
	normalize	28.2%	2.20%	**33.7%**	3.40%	35.1%	2.80%	**44.4%**	5.80%	36.6%	3.80%	**48.2%**	6.00%
	+ center	26.6%	2.10%	32.8%	2.90%	32.9%	2.70%	42.0%	4.50%	34.6%	3.50%	43.9%	5.50%
rev	vanilla	**53.8%**	11.85%	51.7%	11.37%	**58.3%**	11.99%	56.6%	12.59%	**74.3%**	23.60%	73.6%	22.30%
	normalize	53.3%	11.61%	50.0%	10.90%	58.0%	12.35%	56.5%	12.59%	73.7%	24.20%	72.8%	22.10%
	+ center	51.7%	11.37%	53.3%	11.14%	57.1%	11.99%	57.7%	12.35%	69.7%	22.20%	73.5%	23.00%

Table 55: Precision@10 of forward and reverse NN translations with and without the orthogonality constraint and related techniques at vocabulary cutoffs 8192 to 32768. `any` and `disamb` are explained in section 8.6.3. The source has been an `AdaGram` model in 800 dimensions, $\alpha = .1$, trained on Webkorpusz with the vocabulary cut off at 8192 sense vectors.

Tomašev N. 2013), and especially in embedding-based dictionary induction (Dinu, Lazaridou, and Baroni 2015; Lazaridou, Dinu, and Baroni 2015) is when there are *hubs*, data points (target words) returned as the NN (translation) of many points (Wxs), resulting in incorrect hits (translations) in most of the cases. Dinu, Lazaridou, and Baroni (2015) attack the problem with a method they call *global correction*. Here, instead of the original NN, which we will call *forward* NN search to contrast with the more sophisticated method, they first rank source words by their similarity to target words. In *reverse* nearest neighbor (rNN) search, source words are translated to the target words to which they have the lowest (forward) NN rank.[14]

In reverse NN search, we restricted the vocabulary to some tens of thousands of the most frequent words. We introduced this restriction for memory saving, because the $|V_{sr}| \times |V_{tg}|$ similarity matrix has to be sorted column-wise for forward and row-wise for reverse ranking, so at some point of the computation we keep the whole integer matrix of forward NN ranks in memory. It turned out that the restriction makes the results better: a vocabulary cutoff of $2^{15} = 32768$ both on the source and the target size yields slightly better results (74.3%) than the more ambitious $2^{16} = 65536$ (73.9%). This is not the case for forward NN search, where accuracy increases with vocabulary limit (but remains far below that of reverse NN).

8.5.2 *Orthogonal restriction and other tricks*

Xing et al. (2015) note that the original linear translation method is theoretically inconsistent due to its being based on three different similarity measures: `word2vec` itself uses the dot-product of unnormalized vectors, the translation is trained based on Euclidean distance, and

14 If more target words have the same forward rank, Dinu, Lazaridou, and Baroni (2015) make the decision based on cosine similarity. This tie breaking has not proven useful in our experiments.

neighbors are queried based on cosine similarity. They make the framework more coherent by length-normalizing the embeddings, and restricting W to preserve vector length: their matrix W is orthogonal, i.e. the mapping is a rotation. Faruqui and Dyer (2014) achieve even better results by mapping the two embeddings to a lower-dimensional bilingual space with canonical correlation analysis. Artetxe, Labaka, and Agirre (2016) analyze elements of these two works both theoretically and empirically, and find a combination that improves upon dictionary generation and also preserves analogies Mikolov (2013) like

$$\textbf{woman} + \textbf{king} - \textbf{man} \approx \textbf{queen}$$

among the mapped points Wx_i. They find that the orthogonality constraint is key to preserve performance in analogies, and it also improves bilingual performance. In their experiments, length normalization, when followed by centering the embeddings to $\mathbf{0}$ mean, obtains further improvements in bilingual performance without hurting monolingual performance.

8.6 EXPERIMENTS

8.6.1 *Data*

We trained `neela`, `AdaGram`[16], and `mutli` models on (original and stemmed[17] forms of) two semi-gigaword (.7–.8 B words) Hungarian corpora, the Hungarian Webcorpus (Webkorpusz, Halácsy et al. (2004)) and (the non-social-media part of) the Hungarian National Corpus (HNC, Oravecz, Váradi, and Sass (2014)). We used Wiktionary as our seed dictionary, extracted with `wikt2dict`[18] (Ács, Pajkossy, and Kornai 2013). We tried several English embeddings as target, including the 300 dimensional skip-gram with negative sampling model `GoogleNews` released with `word2vec` (Mikolov, Chen, et al. 2013)[19], and those released with GloVe (Pennington, Socher, and Manning 2014)[20]. We report the best results, which were obtained with the release GloVe embeddings trained on 840 B words in 300 dimensions.

8.6.2 *Orthogonal constraint*

We implemented the orthogonal restriction by computing the singular value decomposition

16 I would like to thank Sergey Bartunov for help with his tool.
17 Follow-up work reported in section 8.6.5 applied a third option in preprocessing.
18 https://github.com/juditacs/wikt2dict
19 https://code.google.com/archive/p/word2vec/
20 https://nlp.stanford.edu/projects/glove/

	s			covg
E	-0.04849	függő	addict, aerial	0.4
S	0.01821	alkotó	constituent, creator	0.5
S	0.05096	előzetes	preliminary, trailer	1.0
S	0.0974	kapcsolat	affair, conjunction, linkage	0.33
I	0.1361	kocsi	coach, carriage	1.0
S	0.136	futó	runner, bishop	1.0
S	0.1518	keresés	quest, scan	0.67
S	0.1574	látvány	outlook, scenery, prospect	0.6
S	0.1626	fogad	bet, greet	1.0
S	0.1873	induló	march, candidate	1.0
I	0.187	nemes	noble, peer	0.67
E	0.1934	eltérés	variance, departure	0.4
E	0.1943	alkalmazás	employ, adaptation	0.33
S	0.2016	szünet	interval, cease, recess	0.43
E	0.2032	kezdeményezés	initiation, initiative	1.0
S	0.2052	zavar	disturbance, annoy, disturb, turmoil	0.57
S	0.2054	megelőző	preceding, preventive	0.29
IE	0.2169	csomó	knot^{I}, lump^{I}, mat^{E}	1.0
E[15]	0.21	remény	outlook, promise, expectancy	0.6
S	0.2206	bemutató	exhibition, presenter	0.67
E	0.2208	egyeztetés	reconciliation, correlation	0.5
S	0.237	előadó	auditorium, lecturer	0.67
E	0.2447	nyilatkozat	profession, declaration	0.4
I	0.2494	gazda	farmer, boss	0.67
I	0.2506	kapu	gate, portal	1.0
I	0.2515	előbbi	anterior, preceding	0.67
I	0.2558	kötelezettség	engagement, obligation	0.67
E	0.265	hangulat	morale, humour	0.5
E	0.2733	követ	succeed, haunt	0.67
SE	0.276	minta	norm^{S}, formula^{E}, specimen^{S}	0.75
S	0.2807	sorozat	suite, serial, succession	1.0
S	0.2935	durva	coarse, gross	0.18
I	0.3038	köt	bind, tie	0.67
E	0.3045	egyezmény	treaty, protocol	0.67
I	0.3097	megkülönböztetés	discrimination, differentiation	0.5
I	0.309	ered	stem, originate	0.5
I	0.319	hirdet	advertise, proclaim	1.0
E	0.3212	tartós	substantial, durable	1.0
I	0.3218	ajánlattevő	bidder, supplier, contractor	0.6
I	0.3299	aláírás	signing, signature	0.67
I	0.333	bír	bear, possess	1.0
I	0.3432	áldozat	sacrifice, victim, casualty	1.0
IE	0.3486	kerület	ward^{I}, borough^{I}, perimeter^{E}	0.3
I	0.3486	utas	fare, passenger	1.0
I	0.3564	szigorú	stern, strict	0.5
I	0.3589	bűnös	sinful, guilty	0.5
I	0.3708	rendes	orderly, ordinary	0.5
I	0.3824	eladó	salesman, vendor	0.5
I	0.3861	enyhe	tender, mild, slight	0.6
I	0.3897	maradék	residue, remainder	0.33
I	0.3986	darab	chunk, fragment	0.4
E	0.4012	hiány	poverty, shortage	0.5
I	0.4093	kutatás	exploration, quest	0.5
⋮				
I	0.4138	tanítás	tuition, lesson	0.67
I	0.4196	őszinte	frank, sincere	0.67
I	0.4229	környék	neighborhood, surroundings, vicinity	0.38
I	0.4446	ítélet	judgement, sentence	0.67
I	0.4501	gyerek	childish, kid	0.67
I	0.4521	csatorna	ditch, sewer	0.4
I	0.4547	felügyelet	surveillance, inspection, supervision	0.43
E	0.4551	ritka	rare, odd	0.5
S	0.4563	szerető	fond, lover, affectionate, mistress	0.67
I	0.4608	szeretet	affection, liking	0.67
I	0.4723	vizsgálat	inquiry, examination	0.67
I	0.4853	tömeg	mob, crowd	0.5
I	0.4903	puszta	pure, plain	0.22
I	0.4904	srác	kid, lad	1.0
I	0.4911	büntetés	penalty, sentence	0.29
I	0.4971	képviselő	delegate, representative	0.67
I	0.4975	határ	boundary, border	0.67
I	0.5001	drága	precious, dear, expensive	1.0
S	0.5093	uralkodó	prince, ruler, sovereign	0.5
I	0.5097	válás	separation, divorce	0.67
I	0.5103	ügyvéd	lawyer, advocate	0.67
I	0.5167	előnyös	advantageous, profitable, favourable	1.0
I	0.5169	merev	rigid, strict	1.0
I	0.5204	nyíltan	openly, outright	1.0
I	0.5217	noha	notwithstanding, albeit	1.0
I	0.5311	hulladék	litter, garbage, rubbish	0.43
I	0.5311	szemét	litter, garbage, rubbish	0.43
I	0.5612	kielégítő	satisfying, satisfactory	1.0
E	0.5617	vicc	joke, humour	1.0
I	0.5737	szállító	supplier, vendor	1.0
I	0.5747	óvoda	nursery, daycare, kindergarten	1.0
I	0.5754	hétköznapi	mundane, everyday, ordinary	0.75
I	0.5797	anya	mum, mummy	1.0
I	0.5824	szomszédos	neighbouring, neighbour	0.4
E	0.5931	szabadság	liberty, independence	1.0
I	0.6086	lelkész	pastor, priest	0.4
I	0.6304	fogalom	notion, conception	1.0
I	0.6474	fizetés	salary, wage	0.67
I	0.6551	táj	landscape, scenery	1.0
I	0.6583	okos	clever, smart	0.67
I	0.6707	autópálya	highway, motorway	0.5
I	0.6722	tilos	prohibited, forbidden	1.0
I	0.6811	bevezető	introduction, introductory	1.0
I	0.7025	szövetség	coalition, alliance, union	0.75
I	0.7065	fáradt	exhausted, tired, weary	1.0
I	0.7066	kiállítás	exhibit, exhibition	0.67
I	0.7135	hirdetés	advert, advertisement	1.0
I	0.7147	ésszerű	rational, logical	1.0
I	0.7664	logikai	logic, logical	1.0
I	0.7757	szervez	organise, organize, arrange	1.0
I	0.8122	furcsa	strange, odd	0.4
I	0.8277	azután	afterwards, afterward	0.67
I	0.8689	megbízható	dependable, reliable	0.67

Table 56: Hungarian words with the rNN@1 translations of their sense vectors. The first column is a post-hoc annotation by András Kornai (E error in translation, I identical, S separate meanings), s is the cosine similarity of the translations, and *covg* denotes the coverage of the @1 translations over all gold (good) translations.

[5] The basic translations *hope* is missing

$$U\Sigma V = S_t^\top T_t$$

where S_t and T_t are the matrices consisting of the embedding vectors of the training word pairs in the source and the target space, respectively, and taking

$$W = U\mathbf{1}V$$

where $\mathbf{1}$ is the rectangular identity matrix of appropriate shape. The orthogonal approximation was implemented following a code[21] by Gábor Borbély.

Table 55 shows the effect of these factors. Precision in forward NN search follows a similar trend to that in (Xing et al. 2015) and Artetxe (2016): the best combination is an orthogonal mapping between length-normalized vectors; however, centering did not help in our experiments. Reverse NNs yield much better results than the simpler method, but none of the orthogonality-related techniques give further improvement here. The cause of reverse NN's apparent insensitivity to length may be the topic of further research.

8.6.3 *Results*

We evaluate MSE models in two ways, referred to as `any` and `disamb`. The method `any` has been used for tuning the (meta)parameters of the source embedding and to choose the target: a traditional, single-sense translation has been trained between the first sense vector of each word form and its translations. (If the training word is ambiguous in the seed dictionary, all translations have been included in the training data.) Exploiting the multiple sense vectors, one word can have more than one translation. During the test, a source word was accepted if `any` of its sense vectors had at least one good translation among its k reverse nearest neighbors (rNN@k).

In `disamb`, we used the same translation matrix as in `any`, and inspected the translations of the different sense vectors to see whether the vectors really model different senses rather than synonyms. The lowest requirement for the non-synonymy of sense vectors s_1, s_2 is that the sets of corresponding good rNN@k translations are different. The ratio of words satisfying this requirement among all words with more than one sense vector is shown as `disamb` in table 57.

The values in Table 57 are low. This can in part be due to that the `neela` and the `mutli` models were trained with lower dimension than the best-performing model, so results here are not comparable among

21 https://github.com/hlt-bme-hu/eval-embed

	dim	α/γ	p	m	any	disamb
HNC	800	.02		100	48.5%	7.6%
`neela` Wk	300	–	2	big	54.0%	12.4%
HNC stem	800	.05		big	55.1%	10.4%
HNC	160	.05	3	200	62.2%	15.0%
`mutli` Wk	300	.25		71	62.9%	17.4%
Webkorpusz	800	.05		100	65.9%	17.4%
HNC	600	.05	5	100	68.6%	16.6%
HNC	600	.1	3	50	69.1%	18.8%
Webkorpusz	800	.1		100	73.9%	23.9%

Table 57: Our measures, `any` and `disamb`, for different MSEs. The source embedding has been trained with `AdaGram`, except for when indicated otherwise (`neela`, `mutli`). The meta-parameters are *dim*ension, the resolution parameter (α in `AdaGram` and γ in `mutli`), the maximum number of *p*rototypes (sense vectors), and the vocabulary cutoff (*m*in-freq, the two models with *big* have practically no cut-off).

these different architectures. Follow-up experiments (conducted after the paper review) are reported in section 8.6.5.

Table 56 shows the successfully disambiguated words sorted by the cosine similarity s of good rNN@1 translations of different sense vectors. (We found that most of the few cases when there are more than two sense vectors with a good rNN@1 translation are due to the fact that the seed dictionary contains some non-basic translation, e.g. *kapcsolat* 'relationship, conjunction' has 'affair' among its seed translations. In these cases, we chose two sense vectors arbitrarily.When there are sense vectors with more than two rNN@k hits , the choice of the corresponding target words is also arbitrary.) Relying on s is similar to the monolingual setting of clustering the sense vectors for each word, but here we restrict our analysis to sense vectors that prove to be sensible in linear translation.

We see that most words with $s < .25$ are really ambiguous from a standard lexicographic point of view, but the translations with $s > .35$ tend to be synonyms instead.

8.6.4 *Part of speech*

The clearest case of homonymy is when unrelated senses belong to different parts of speech (POSs), and the translations reflect these POSs, e.g. *nő* 'woman; increase' or *vár* 'wait; castle'.[22] In purely semantic approaches, like `4lang`(see Chapter 3), POS-difference alone is not enough for analyzing a word as ambiguous, e.g. we see the only difference be-

22 We note that some POSs in Hungarian have blurred borders, e.g. it is debatable whether the nominal *önkéntes* 'voluntary; volunteer' is ambiguous for its POS.

	any	disamb
`AdaGram`	73.3%	18.53%
`mutli` sense vectors	71.0%	19.46%
`mutli` context vectors	69.9%	20.76%

Table 58: The resolution trade-off between translation precision and sense distinctiveness. The source models are 600-dimensional Hungarian models trained on the de-glutinized version of the Hungarian National Corpus. Other meta-parameters have been set to default.

tween the noun and participle senses of *alkalmazott*, 'employee; applied' as *employ*ment being the *appli*cation of people for work; in the case of *belső* 'internal; interior', the noun refers to the part of a building described by the adjective.

More interesting are word forms with related senses in the same POS, e.g. *cikk*, 'item; article' (an article is an item in a newspaper); *eredmény*, 'score; result' (a score is a result measured by a number); *magas*, 'tall; high' (tall is used for people rather than high); or *idegen*, 'strange, alien; foreign', where the English translations are special cases of 'unfamiliar' (person versus language).

8.6.5 *Comparison of `AdaGram` and `mutli`*

After the compilation of Makrai and Lipp (2018), we trained models that enable a more fair comparison of `AdaGram` and `mutli` in terms of semantic resolution: we trained 600-dimensional models for Hungarian to have the 2:1 ratio between the source and the target dimension that has been reported to be optimal for this task (Mikolov, Le, and Sutskever 2013; Makrai 2016a). This time we used the de-glutinized version (Borbély, Kornai, et al. 2016; Nemeskey 2017) of the Hungarian National corpus for better morphological generalization. The word embeddings are available online[23].

We can see in table 58 that there is a trade-off between the two measures, which may be interpreted to indicate that the more specific a vector is, the easier it is to translate, but if the vectors are too specific, then the translations may coincide. [24]

As a direction for future research, the analysis of the observed and inferred number of word senses as a function of word frequency may shed more light on how good a model of word ambiguity the Dirichlet Process is.

23 https://hlt.bme.hu/en/publ/makrai17

24 There are two `mutli` models because Skip-gram and the related MSE models represent each word with two vectors, u and v in the formula $p(w_i \mid w_j) \propto \exp(u_i^\top v_j)$, that `mutli` calls *sense* versus *context* vectors, respectively.

BIBLIOGRAPHY

Abend, Omri, and Ari Rappoport. 2013. "UCCA: A semantics-based grammatical annotation scheme." In *IWCS'13,* 1–12. (Cited on page 57).

———. 2017. "The state of the art in semantic representation." In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* 77–89. (Cited on pages 55, 56).

Ács, Judit, Dániel Lévai, Dávid Márk Nemeskey, and András Kornai. 2021. "Evaluating Contextualized Language Models for Hungarian." In *XVII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2020).* Szeged. (Cited on page 106).

Ács, Judit, Dávid Márk Nemeskey, and Gábor Recski. 2017. "Building word embeddings from dictionary definitions." In *K + K = 120: Papers dedicated to László Kálmán and András Kornai on the occasion of their 60th birthdays,* edited by Katalin Mády Beáta Gyuris and Gábor Recski. Research Institute for Linguistics, Hungarian Academy of Sciences (RIL HAS). (Cited on page 65).

Ács, Judit, Katalin Pajkossy, and András Kornai. 2013. "Building basic vocabulary across 40 languages." In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora,* 52–58. Sofia, Bulgaria: Association for Computational Linguistics. (Cited on pages 62, 66, 143, 201, 212).

Akbik, Alan, Duncan Blythe, and Roland Vollgraf. 2018. "Contextual String Embeddings for Sequence Labeling." In *Proceedings of the 27th International Conference on Computational Linguistics,* 1638–1649. Santa Fe, New Mexico, USA: Association for Computational Linguistics, August. https://www.aclweb.org/anthology/C18-1139. (Cited on page 115).

Allen, James, and Choh Man Teng. 2018. "Putting semantics into semantic roles." In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics.* (Cited on page 169).

Amrami, Asaf, and Yoav Goldberg. 2018. "Word Sense Induction with Neural biLM and Symmetric Patterns." Http://arxiv.org/abs/1808.08518v2, August 26. arXiv: http://arxiv.org/abs/1808.08518v2 [cs.CL]. http://arxiv.org/abs/1808.08518v2. (Cited on page 183).

Anandkumar, Animashree, Rong Ge, Daniel Hsu, Sham M Kakade, and Matus Telgarsky. 2014. "Tensor decompositions for learning latent variable models." *The Journal of Machine Learning Research* 15 (1): 2773–2832. (Cited on page 184).

Andrews, Avery. 2015. "Reconciling NSM and formal semantics." *ms:* v2, jan 2015. (Cited on page 66).

Antoniak, Maria, and David Mimno. 2018. "Evaluating the stability of embedding-based word similarities." *Transactions of the Association for Computational Linguistics* 6:107–119. (Cited on page 80).

Arora, Sanjeev, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2015. "Random Walks on Context Spaces: Towards an Explanation of the Mysteries of Semantic Word Embeddings." *arXiv:1502.03520v1* 4:385–399. (Cited on page 95).

———. 2016. "Linear Algebraic Structure of Word Senses, with Applications to Polysemy." *arXiv:1601.03764v1.* (Cited on pages 98, 99, 157).

Arora, Sanjeev, Yingyu Liang, and Tengyu Ma. 2017. "A simple but tough-to-beat baseline for sentence embeddings." In *International Conference on Learning Representations.* https://openreview.net/pdf?id=SyK00v5xx. (Cited on page 133).

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre. 2016. "Learning principled bilingual mappings of word embeddings while preserving monolingual invariance." In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing.* (Cited on pages 212, 214).

Athiwaratkun, Ben, Andrew Gordon Wilson, and Anima Anandkumar. 2018. "Probabilistic FastText for Multi-Sense Word Embeddings." *arXiv preprint arXiv:1806.02901.* (Cited on page 184).

Avraham, Oded, and Yoav Goldberg. 2016. "Improving reliability of word similarity evaluation by redesigning annotation task and performance measure." *arXiv preprint arXiv:1611.03641.* (Cited on page 99).

———. 2017. "The interplay of semantics and morphology in word embeddings," *arXiv preprint arXiv:1704.01938.* (Cited on page 105).

Bailey, Eric, Charles Meyer, and Shuchin Aeron. 2018. "Learning semantic word respresentations via tensor factorization." ArXiv:1704.02686. https://openreview.net/forum?id=B1kIr-WRb. (Cited on pages 176, 178, 179, 182, 184, 186, 194).

Baker, Collin F., Charles J. Fillmore, and John B. Lowe. 1998. "The Berkeley FrameNet Project." In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1,* 86–90. ACL '98. Montreal, Quebec, Canada: Association for Computational Linguistics. doi:10.3115/980845.980860. http://dx.doi.org/10.3115/980845.980860. (Cited on page 43).

Balažević, Ivana, Carl Allen, and Timothy M Hospedales. 2019. "TuckER: Tensor Factorization for Knowledge Graph Completion." ArXiv preprint arXiv:1901.09590. (Cited on page 183).

Banarescu, Laura, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. "Abstract Meaning Representation for Sembanking." In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse,* 178–186. Sofia, Bulgaria: Association for Computational Linguistics. https://www.aclweb.org/anthology/W13-2322. (Cited on pages 50, 60).

Baroni, M., S. Bernardini, A. Ferraresi, and E. Zanchetta. 2009. "The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora." In *LREC 2009,* 3:209–226. (Cited on page 201).

Baroni, M., G. Dinu, and G. Kruszewski. 2014. "Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors." In *Proceedings of ACL 2014,* 237–247. (Cited on page 97).

Baroni, Marco, and Alessandro Lenci. 2010. "Distributional memory: A general framework for corpus-based semantics." *Computational Linguistics* 36 (4): 673–721. (Cited on page 20).

Bartunov, Sergey, Dmitry Kondrashkin, Anton Osokin, and Dmitry Vetrov. 2016. "Breaking Sticks and Ambiguities with Adaptive Skip-gram," *Proceedings of Machine Learning Research* 51: Artificial Intelligence and Statistics (May): 130–138. (Cited on pages 183, 209).

Belinkov, Yonatan, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017a. "What do neural machine translation models learn about morphology?" In *ACL.* ArXiv preprint arXiv:1704.03471. (Cited on page 105).

———. 2017b. "What do Neural Machine Translation Models Learn about Morphology?" In *Proc. of ACL.* doi:10.18653/v1/P17-1080. https://www.aclweb.org/anthology/P17-1080. (Cited on page 128).

Bengio, Yoshua, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. "A Neural Probabilistic Language Model." *Journal of Machine Learning Research* 3:1137–1155. http://www.jmlr.org/papers/v3/bengio03a.html. (Cited on pages 78, 91, 92).

Berend, Gábor. 2017. "Sparse Coding of Neural Word Embeddings for Multilingual Sequence Labeling." *Transactions of the Association for Computational Linguistics* 5:247–261. ISSN: 2307-387X. https://transacl.org/ojs/index.php/tacl/article/view/1063. (Cited on pages 157, 159, 160).

Berend, Gábor, Márton Makrai, and Péter Földiák. 2018. "300-sparsans at SemEval-2018 Task 9: Hypernymy as interaction of sparse attributes." In *Proceedings of The 12th International Workshop on Semantic Evaluation,* 928–934. New Orleans, Louisiana: Association for Computational Linguistics, June. doi:10.18653/v1/S18-1152. https://aclanthology.org/S18-1152. (Cited on pages 157, 158).

Blei, David M, Andrew Y Ng, and Michael I Jordan. 2003. "Latent dirichlet allocation." *Journal of machine Learning research* 3 (Jan): 993–1022. (Cited on page 81).

Boguraev, Branimir K., and Edward J. Briscoe. 1989. *Computational Lexicography for Natural Language Processing.* Longman. (Cited on pages 39–41, 208).

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. "Enriching Word Vectors with Subword Information." *Transactions of the Association for Computational Linguistics* 5:135–146. ISSN: 2307-387X. https://transacl.org/ojs/index.php/tacl/article/view/999. (Cited on pages 92, 103, 104).

Bojanowski, Piotr, Armand Joulin, and Tomas Mikolov. 2016. "Alternative Structures for Character-level RNNs." In *International Conference on Learning Representations, Workshop track (ICLR 2016).* arXiv: 1511.06303 [cs.LG]. (Cited on page 109).

Borbély, Gábor, András Kornai, Dávid Nemeskey, and Marcus Kracht. 2016. "Denoising composition in distributional semantics." In *DSALT: Distributional Semantics and Linguistic Theory.* Poster. (Cited on pages 106, 216).

Borbély, Gábor, Márton Makrai, Dávid Márk Nemeskey, and András Kornai. 2016. "Evaluating multi-sense embeddings for semantic resolution monolingually and in word translation." In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP,* 83–89. Berlin, Germany: Association for Computational Linguistics. doi:10.18653/v1/W16-2515. http://www.aclweb.org/anthology/W16-2515. (Cited on pages 183, 198, 207, 208, 210).

Botha, Jan A, and Phil Blunsom. 2014. "Compositional Morphology for Word Representations and Language Modelling." In *ICML,* 1899–1907. (Cited on page 105).

Bouma, Gerlof. 2009. "Normalized (pointwise) mutual information in collocation extraction." In *GSCL 2009: International Conference of the German Society for Computational Linguistics and Language Technology.* (Cited on pages 175, 178).

Brachman, R.J., and H. Levesque. 1985. *Readings in knowledge representation.* Morgan Kaufmann Publishers Inc., Los Altos, CA. (Cited on pages 15, 37, 39).

Brants, Thorsten, and Alex Franz. 2006. *Web 1T 5-gram Version 1.* Philadelphia: Linguistic Data Consortium. (Cited on page 66).

Bresnan, Joan. 1978. "A realistic transformational grammar." In *Linguistic theory and psychological reality,* edited by M. Halle, J. Bresnan, and G.A. Miller. MIT Press. (Cited on page 34).

———. 2001. *Lexical-Functional Syntax.* Oxford, UK: Blackwell. (Cited on page 34).

Brown, P.F., V.J. Della Pietra, P.V. de Souza, J.C. Lai, and R.L. Mercer. 1992. "Class–based n–gram models of natural language." *Computational Linguistics* 18 (4): 467–480. (Cited on page 91).

Bullon, Stephen. 2003. *Longman Dictionary of Contemporary English.* 4th ed. Longman. (Cited on page 61).

Butt, Miriam. 2006. *Theories of Case.* Cambridge University Press. (Cited on page 60).

Camacho-Collados, Jose, Claudio Delli Bovi, Luis Espinosa-Anke, Sergio Oramas, Tommaso Pasini, Enrico Santus, Vered Shwartz, Roberto Navigli, and Horacio Saggion. 2018. "SemEval-2018 Task 9: Hypernym Discovery." In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018).* New Orleans, LA, United States: Association for Computational Linguistics. (Cited on page 158).

Camacho-Collados, José, and Roberto Navigli. 2016. "Find the word that does not belong: A framework for an intrinsic evaluation of word vector representations." In *Proceedings of the ACL Workshop on Evaluating Vector Space Representations for NLP,* 43–50. (Cited on page 186).

Cao, Kris, and Marek Rei. 2016. "A Joint Model for Word Embedding and Word Morphology." In *Repl4NLP.* arXiv: 1606.02601. http://arxiv.org/abs/1606.02601. (Cited on page 105).

Carroll, J. D., and J. J. Chang. 1970. "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition." *Psychometrika* 35:283–319. (Cited on page 180).

Chafe, W. L. 1970. *Meaning and the structure of language.* Chicago: University Press. (Cited on page 173).

Chomsky, Noam. 1965. *Aspects of the Theory of Syntax.* MIT Press. (Cited on page 24).

———. 1970. "Remarks on nominalization." In *Readings in English Transformational Grammar,* edited by R. Jacobs and P. Rosenbaum, 184–221. Waltham, MA: Blaisdell. (Cited on page 60).

Church, Kenneth W., and Patrick Hanks. 1990. "Word association norms, mutual information, and lexicography." *Computational Linguistics* 16 (1): 22–29. (Cited on page 79).

Cilibrasi, Rudi, and Paul Vitányi. 2004. *Automatic Meaning Discovery Using Google.* (Cited on page 84).

Cimiano, Philipp, Andreas Hotho, and Steffen Staab. 2005. "Learning concept hierarchies from text corpora using formal concept analysis." *Journal Artificial Intelligence Research (JAIR)* 24 (1): 305–339. (Cited on page 158).

Coenen, Andy, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. 2019. "Visualizing and Measuring the Geometry of BERT." *arXiv preprint arXiv:1906.02715.* (Cited on pages 127, 128).

Collados, José Camacho, Claudio Delli Bovi, Alessandro Raganato, and Roberto Navigli. 2016. "A Large-Scale Multilingual Disambiguation of Glosses." Http://arxiv.org/abs/1608.06718v1, August 24. arXiv: `http://arxiv.org/abs/1608.06718v1 [cs.CL]`. `http://arxiv.org/abs/1608.06718v1`. (Cited on page 181).

Collins, A.M., and E.F. Loftus. 1975. "A spreading-activation theory of semantic processing." *Psychological Review* 82:407–428. (Cited on pages 7–9, 11, 65).

Collobert, R., J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. 2011. "Natural Language Processing (Almost) from Scratch." *Journal of Machine Learning Research (JMLR).* (Cited on pages xii, 92, 93, 150, 154).

Collobert, Ronan, and Jason Weston. 2008. "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning." In *Proceedings of the 25th International Conference on Machine Learning,* 160–167. ICML '08. Helsinki, Finland: ACM. (Cited on page 138).

Conneau, Alexis, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. "What you can cram into a single \$&!#* vector: Probing sentence embeddings for linguistic properties." In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* 2126–2136. Melbourne, Australia: Association for Computational Linguistics. http://aclweb.org/anthology/P18-1198. (Cited on page 128).

Cotterell, Ryan, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Arya D. McCarthy, Katharina Kann, et al. 2018. "The CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection." In *Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection.* Brussels, Belgium: Association for Computational Linguistics, October. (Cited on page 105).

Cotterell, Ryan, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. "The SIGMORPHON 2016 Shared Task—Morphological Reinflection." In *Proceedings of the 2016 Meeting of SIGMORPHON.* Berlin, Germany: Association for Computational Linguistics, August. (Cited on page 105).

Cotterell, Ryan, and Hinrich Schütze. 2015. "Morphological word-embeddings." In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 1287–1292. (Cited on page 105).

Dahl, George E, Dong Yu, Li Deng, and Alex Acero. 2011. "Large vocabulary continuous speech recognition with context-dependent DBN-HMMs." In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on,* 4688–4691. IEEE. (Cited on pages 115, 137).

Dalvi, Fahim, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. 2019. "What is one grain of sand in the desert? analyzing individual neurons in deep nlp models." In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI).* (Cited on page 106).

Dalvi, Fahim, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, and Stephan Vogel. 2017. "Understanding and improving morphological learning in the neural machine translation decoder." In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers),* 1:142–151. (Cited on page 105).

Deerwester, Scott C., Susan T Dumais, and Richard A. Harshman. 1990. "Indexing by latent semantic analysis." *Journal of the American Society for Information Science* 41 (6): 391–407. (Cited on pages 78, 98, 99).

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." *arXiv preprint arXiv:1810.04805* (October 11). arXiv: 1810.04805v1 [cs.CL]. http://arxiv.org/abs/1810.04805v1. (Cited on pages 116, 183).

———. 2019. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In *Proc. of NAACL.* (Cited on page 116).

Diederich, Paul Bernard. 1939. *The frequency of Latin words and their endings.* The University of Chicago Press. (Cited on page 66).

Dinu, Georgiana, Angeliki Lazaridou, and Marco Baroni. 2015. "Improving Zero-shot Learning by Mitigating the Hubness Problem." ICLR 2015, Workshop Track. arXiv: 1412.6568 [cs.CL]. (Cited on pages 114, 201, 211).

Döbrössy, Bálint, Márton Makrai, Balázs Tarján, and György Szaszák. 2019. "Investigating Sub-Word Embedding Strategies for the Morphologically Rich and Free Phrase-Order Hungarian." In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019),* 187–193. Florence, Italy: Association for Computational Linguistics, August. doi:10.18653/v1/W19-4321. https://www.aclweb.org/anthology/W19-4321. (Cited on pages 78, 103).

Domingos, Pedro. 2012. "A few useful things to know about machine learning." In *Communications of the ACM,* 55:78–87. ACM New York, NY, USA, October. (Cited on page 155).

Dowty, David. 1991. "Thematic Proto-Roles and Argument Selection." *Language* 67 (3): 547–619. (Cited on page 45).

Dressler, Wolfgang U, and Mária Ladányi. 2000. "Productivity in word formation (WF): a morphological approach." *Acta Linguistica Hungarica* 47 (1-4): 103–145. (Cited on page 186).

Dumais, Susan T, George W Furnas, Thomas K Landauer, Scott Deerwester, and Richard Harshman. 1988. "Using latent semantic analysis to improve access to textual information." In *Proceedings of the SIGCHI conference on Human factors in computing systems,* 281–285. (Cited on page 80).

Enarvi, Seppo, Peter Smit, Sami Virpioja, Mikko Kurimo, Seppo Enarvi, Peter Smit, Sami Virpioja, and Mikko Kurimo. 2017. "Automatic speech recognition with very large conversational Finnish and Estonian vocabularies." *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* 25 (11): 2085–2097. (Cited on page 106).

Endres, Dominik, Peter Földiák, and Uta Priss. 2010. "An Application of Formal Concept Analysis to Semantic Neural Decoding." Reviewed, *Annals of Mathematics and Artificial Intelligence* 57, nos. 3-4 (July): 233–248. doi:10.1007/s10472-010-9196-8. (Cited on pages 158, 160).

Ettinger, Allyson. 2020. "What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models." *Transactions of the Association for Computational Linguistics* 8:34–48. (Cited on pages 117, 130–132).

Faruqui, Manaal, Jesse Dodge, Sujay Jauhar, Chris Dyer, Ed Hovy, and Noah Smith. 2015. "Retrofitting Word Vectors to Semantic Lexicons." In *Proceedings of NAACL 2015.* Best Student Paper Award. (Cited on pages 46, 100, 157).

Faruqui, Manaal, and Chris Dyer. 2014. "Improving Vector Space Word Representations Using Multilingual Correlation." In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics,* 462–471. Gothenburg, Sweden: Association for Computational Linguistics. doi:10.3115/v1/E14-1049. http://www.aclweb.org/anthology/E14-1049. (Cited on pages 101, 212).

Faruqui, Manaal, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. 2016. "Problems With Evaluation of Word Embeddings Using Word Similarity Tasks." (Cited on page 114).

Fillmore, Charles. 1968. "The case for case." In *Universals in Linguistic Theory,* edited by E. Bach and R. Harms, 1–90. New York: Holt / Rinehart. (Cited on pages 21, 22, 29, 168, 172).

———. 1977. "The case for case reopened." In *Grammatical Relations,* edited by P. Cole and J.M. Sadock, 59–82. Academic Press. (Cited on page 152).

Findler, Nicholas V., ed. 1979. *Associative Networks: Representation and Use of Knowledge by Computers.* Academic Press. (Cited on page 1).

Firth, John R. 1957. "A synopsis of linguistic theory." In *Studies in linguistic analysis,* 1–32. Blackwell. (Cited on page 78).

Frandsen, Abraham, and Rong Ge. 2019. "Understanding composition of word embeddings via tensor decomposition." In *7th International Conference on Learning Representations, ICLR 2019.* ArXiv preprint arXiv:1902.00613. May. https://openreview.net/forum?id=H1eqjiCctX. (Cited on pages 176, 185, 186).

Fried, Daniel, Tamara Polajnar, and Stephen Clark. 2015. "Low-Rank Tensors for Verbs in Compositional Distributional Semantics." In *ACL.* (Cited on page 176).

Furnas, Susan T Dumais George W, Thomas K Landauer, Scott Deerwester, and Richard Harshman. 1988. "Using latent semantic analysis to improve access to textual information." In *Proceedings of the SIGCHI conference on Human factors in computing systems.* Citeseer. (Cited on page 80).

Fyshe, Alona, Leila Wehbe, Partha P Talukdar, Brian Murphy, and Tom M Mitchell. 2015. "A compositional and interpretable semantic space." In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 32–41. (Cited on page 157).

Ganter, Bernhard, and Rudolf Wille. 2012. *Formal concept analysis: mathematical foundations.* Springer Science & Business Media. (Cited on page 158).

Gittens, Alex, Dimitris Achlioptas, and Michael W. Mahoney. 2017. "Skip-Gram – Zipf + Uniform = Vector Additivity." In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* 69–76. Vancouver, Canada: Association for Computational Linguistics. doi:10.18653/v1/P17-1007. http://aclweb.org/anthology/P17-1007. (Cited on page 177).

Gladkova, Anna, and Aleksandr Drozd. 2016. "Intrinsic Evaluations of Word Embeddings: What Can We Do Better?" In *Proc. RepEval (this volume),* edited by Omer Levy. ACL. (Cited on pages 103, 108, 110, 111, 207).

Glavaš, Goran, and Ivan Vulić. 2018. "Explicit Retrofitting of Distributional Word Vectors." In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* 34–45. Melbourne, Australia: Association for Computational Linguistics. doi:10.18653/v1/P18-1004. https://www.aclweb.org/anthology/P18-1004. (Cited on page 102).

Goddard, Cliff. 2002. "The search for the shared semantic core of all languages." In *Meaning and Universal Grammar – Theory and Empirical Findings,* edited by Cliff Goddard and Anna Wierzbicka, 1:5–40. Benjamins. (Cited on page 60).

Goddard, Cliff, and Anna Wierzbicka, eds. 1994. *Semantic and Lexical Universals.* John Benjamins Publishing Company. (Cited on page 22).

Goldberg, Adele. 2006. *Constructions at Work: The Nature of Generalization in Language.* Oxford University Press, USA. (Cited on page 122).

Goldberg, Yoav, and Omer Levy. 2014. "word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method." *arXiv preprint arXiv:1402.3722.* (Cited on page 94).

Gove, Philip Babcock, ed. 1961. *Webster's Third New International Dictionary of the English Language, Unabridged.* G. & C. Merriam. (Cited on page 61).

Grefenstette, Edward, and Mehrnoosh Sadrzadeh. 2011. "Experimenting with transitive verbs in a DisCoCat." In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics,* 62–66. Edinburgh, UK: Association for Computational Linguistics, July. https://www.aclweb.org/anthology/W11-2507. (Cited on pages x, 186).

Grice, Paul, and Peter Strawson. 1956. "In defense of a dogma." *The Philosophical Review* 65:148–152. (Cited on page 72).

Gruber, Jeffrey Steven. 1965. "Studies in lexical relations." PhD diss., Massachusetts Institute of Technology. (Cited on page 27).

Gruber, Thomas R, et al. 1993. "A translation approach to portable ontology specifications." *Knowledge acquisition* 5 (2): 199–220. (Cited on page 72).

Halácsy, Péter, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. 2004. "Creating open language resources for Hungarian." In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004),* 203–210. ELRA. (Cited on pages 66, 141, 143, 147, 201, 212).

Han, Lejia, and John C. Bancroft. 2010. "Nearest approaches to multiple lines in n-dimensional space." In *CREWES Research Report,* vol. 22. University of Calgary. (Cited on page 155).

Harris, Randy Allen. 1995. *The Linguistics Wars.* Oxford University Press.

Harris, Zellig. 1951. *Methods in Structural Linguistics.* University of Chicago Press. (Cited on page 78).

Harris, Zellig S. 1954. "Distributional structure." *Word* 10 (23): 146–162. (Cited on page 98).

Harshman, R. A. 1970. "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis." *UCLA Working Papers in Phonetics* 16:1–84. http://publish.uwo.ca/~harshman/wpppfac0.pdf. (Cited on page 180).

Hashimoto, Kazuma, Pontus Stenetorp, Makoto Miwa, and Yoshimasa Tsuruoka. 2014. "Jointly Learning Word Representations and Composition Functions Using Predicate-Argument Structures." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing,* 1544–1555. (Cited on page 188).

Hashimoto, Kazuma, and Yoshimasa Tsuruoka. 2015. "Learning embeddings for transitive verb disambiguation by implicit tensor factorization." In *3rd Workshop on Continuous Vector Space Models and their Compositionality.* (Cited on page 176).

Hausser, Roland. 1984. *Surface compositional grammar.* München: Wilhelm Fink Verlag. (Cited on page 154).

Hayes, Patrick J. 1979. "The naive physics manifesto." In *Expert Systems in the Micro-Electronic Age,* edited by D. Michie, 242–270. Edinburgh University Press. (Cited on pages 16, 48, 49).

Hays, David G. 1964. "Dependency theory: a formalism and some observations." *Language* 40(4):511–525. (Cited on page 7).

Héja, Enikő, and Dávid Takács. 2012. "An Online Dictionary Browser for Automatically Generated Bilingual Dictionaries." In *Proceedings of EURALEX2012,* 468–477. (Cited on page 143).

Heringer, T. 1967. "Wertigkeiten und nullwertige Verben im Deutschen." *Zeitschrift für Deutsche Spache* 23:13–34. (Cited on page 172).

Hewitt, John, and Christopher D Manning. 2019. "A structural probe for finding syntax in word representations." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers),* 4129–4138. (Cited on page 128).

Hill, Felix, Roi Reichart, and Anna Korhonen. 2014. "Simlex-999: Evaluating semantic models with (genuine) similarity estimation." *Computational Linguistics* 41 (4): 665–695. (Cited on page 101).

———. 2015. "Simlex-999: Evaluating semantic models with (genuine) similarity estimation." *Computational Linguistics* 41 (4): 665–695. (Cited on page 99).

Hinton, Geoffrey Everest, James Lloyd McClelland, and David Everett Rumelhart. 1986. "Distributed representations." Chap. 3 in *Parallel distributed processing: Explorations in the microstructure of cognition,* edited by James Lloyd McClelland and David Everett Rumelhart, 1:77–109. Cambridge, MA: MIT Press. (Cited on page 89).

Hobbs, J.R. 2008. "Deep Lexical Semantics." *Lecture Notes in Computer Science* 4919:183. (Cited on pages 16, 46, 48, 49).

Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. "Long Short-Term Memory." *Neural Computation* 9, no. 8 (November): 1735–1780. (Cited on page x).

Howard, Jeremy, and Sebastian Ruder. 2018. "Universal Language Model Fine-tuning for Text Classification." In *ACL.* (Cited on pages 115, 183).

Huang, Eric, Richard Socher, Christopher Manning, and Andrew Ng. 2012. "Improving Word Representations via Global Context and Multiple Word Prototypes." In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012),* 873–882. Jeju Island, Korea: Association for Computational Linguistics. (Cited on pages 101, 154, 183, 209).

Iliev, R, M Dehghani, and E Sagi. 2014. "Automated text analysis in psychology: Methods, applications, and future developments." *Language and Cognition.* (Cited on page 80).

Ittzés, Nóra, ed. 2011. *A magyar nyelv nagyszótára III-IV.* Akadémiai Kiadó. (Cited on page 209).

Jackendoff, Ray S. 1972. *Semantic Interpretation in Generative Grammar.* MIT Press. (Cited on page 27).

———. 1983. *Semantics and Cognition.* MIT Press. (Cited on page 27).

———. 1990. *Semantic Structures.* MIT Press. (Cited on pages 20, 27, 28, 30, 31).

Jastrzebski, Stanisław, Leśniak, Damian, Czarnecki, and Wojciech Marian. 2017. "How to evaluate word embeddings? on importance of data efficiency and simple supervised tasks." *arXiv preprint arXiv:1702.02170.* (Cited on page 185).

Jenatton, Rodolphe, Nicolas Le Roux, Antoine Bordes, and Guillaume Obozinski. 2012. "A Latent Factor Model for Highly Multi-relational Data." In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2,* 3167–3175. NIPS'12. Lake Tahoe, Nevada, USA: Curran Associates Inc. http://dl.acm.org/citation.cfm?id=2999325.2999488. (Cited on pages 178, 182, 183, 186).

Joulin, Armand, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. "Bag of Tricks for Efficient Text Classification." ArXiv preprint arXiv:1607.01759. (Cited on pages 105, 106).

Jurafsky, Dan. 2014. *Charles J. Fillmore.* (Cited on pages 42, 43).

Kalivoda, Ágnes. 2019. "Véges erőforrás végtelen sok igekötős igére [A finite resource for infinitely many Hungarian particle verbs]." In *XV. Magyar Számítógépes Nyelvészeti Konferencia,* edited by Gábor Berend, Gábor Gosztolya, and Veronika Vincze, 331–344. January. (Cited on page 186).

Kalivoda, Ágnes. 2021. "Igekötős szerkezetek a magyarban." PhD diss., Pázmány Péter Katolikus Egyetem, Bölcsészet- és Társadalomtudományi Kar, Nyelvtudományi Doktori Iskola. (Cited on page 196).

Kalivoda, Ágnes, Noémi Vadász, and Balázs Indig. 2018. "Manócska: A Unified Verb Frame Database for Hungarian." In *Proceedings of the 21st International Conference on Text, Speech and Dialogue—TSD 2018, Brno, Czech Republic,* edited by Petr Sojka, Aleš Horák, Ivan Kopeček, and Karel Pala, 11107:135–143. Lecture Notes in Artificial Intelligence. Springer-Verlag, September. ISBN: 978-3-030-00794-2. doi:https://doi.org/10.1007/978-3-030-00794-2_14. (Cited on page 186).

Kartsaklis, Dimitri, and Mehrnoosh Sadrzadeh. 2014. "A Study of Entanglement in a Categorical Framework of Natural Language." In *The 11th workshop on Quantum Physics and Logic.* ArXiv:1412.8102. June. (Cited on pages x, 186–188).

Katz, J., and Jerry A. Fodor. 1963. "The structure of a semantic theory." *Language* 39:170–210. (Cited on pages 19–21).

Katz, Jerrold J. 1987. "Common Sense in Semantics." In *New Directions in Semantics,* edited by E. Lepore, 157–234. Academic Press. (Cited on page 23).

Kazemi, Seyed Mehran, and David Poole. 2018. "SimplE Embedding for Link Prediction in Knowledge Graphs." In *NIPS.* (Cited on page 183).

Kennedy, Alistair, and Stan Szpakowicz. 2008. "Evaluating Roget's Thesauri." In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL-2008),* 416–424. The Association for Computer Linguistics, June. (Cited on page 36).

———. 2014. "Evaluation of automatic updates of Roget's Thesaurus." *Journal of Language Modelling* 2 (1): 1–49. (Cited on pages 36, 37).

Kilgarriff, Adam, Pavel Rychlý, Pavel Smrž, and David Tugwell. 2004. "Sketch engine." In *Proceedings of Euralex,* edited by Geoffrey Williams and Sandra Vessier, 105–116. Lorient, Université de Bretagne-Sud, Faculté des lettres et des sciences humaines, July. (Cited on page 178).

Kipper, Karin, Anna Korhonen, Neville Ryant, and Martha Palmer. 2008. "A large-scale classification of English verbs." *Language Resources and Evaluation* 42 (1): 21–40. (Cited on pages 44, 195).

Kolda, Tamara G, and Brett W Bader. 2009. "Tensor decompositions and applications." *SIAM review* 51 (3): 455–500. (Cited on pages 176, 180, 181).

Koller, Alexander, Stephan Oepen, and Weiwei Sun. 2019. "Graph-based meaning representations: Design and processing." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts,* 6–11. (Cited on page 58).

Komlósy, András. 1982. "Deep structure cases reinterpreted." In *Hungarian General Linguistics,* edited by Ferenc Kiefer, 351–385. John Benjamins. Amsterdam–Philadelphia. (Cited on page 172).

Kornai, András. 2008. *Mathematical Linguistics.* Advanced Information and Knowledge Processing. Springer. ISBN: 9781846289859. (Cited on pages 42, 62).

———. 2011. "Probabilistic grammars and languages." *Journal of Logic, Language, and Information* 20:317–328. (Cited on page 19).

———. 2012. "Eliminating ditransitives." In *Revised and Selected Papers from the 15th and 16th Formal Grammar Conferences,* edited by Ph. de Groote and M-J Nederhof, 243–261. LNCS 7395. Springer. (Cited on pages 64, 68, 152).

———. 2019. *Semantics.* Springer Verlag. ISBN: 978-3-319-65644-1. http://kornai.com/Drafts/sem.pdf. (Cited on pages 21, 36, 62, 64, 67, 72, 75, 79).

———. 2022. *Vector semantics.* Springer Verlag. http://kornai.com/Drafts/advsem.pdf. (Cited on pages 15, 60, 67, 76, 154, 167).

Kornai, András, Judit Ács, Márton Makrai, Dávid Márk Nemeskey, Katalin Pajkossy, and Gábor Recski. 2015. "Competence in lexical semantics." In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics,* 165–175. Denver, Colorado: Association for Computational Linguistics. doi:10.18653/v1/S15-1019. https://www.aclweb.org/anthology/S15-1019. (Cited on pages 65, 66, 73).

Kornai, András, Péter Halácsy, Viktor Nagy, Csaba Oravecz, Viktor Trón, and Dániel Varga. 2006. "Web-based frequency dictionaries for medium density languages." In *Proc. 2nd Web as Corpus Workshop (EACL 2006 WS01),* edited by A. Kilgariff and M. Baroni, 1–8. (Cited on page 66).

Kornai, András, and Márton Makrai. 2013. "A 4lang fogalmi szótár." In *IX. Magyar Számitógépes Nyelvészeti Konferencia,* edited by Attila Tanács and Veronika Vincze, 62–70. (Cited on pages 61, 65, 70, 152).

Kornai, András, Dávid Márk Nemeskey, and Gábor Recski. 2016. "Detecting Optional Arguments of Verbs." In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016),* edited by Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, et al., 2815–2818. Portorož, Slovenia: European Language Resources Association (ELRA), May. ISBN: 978-2-9517408-9-1. (Cited on page 186).

Kossaifi, Jean, Yannis Panagakis, Animashree Anandkumar, and Maja Pantic. 2016. "Tensorly: Tensor learning in python." ArXiv preprint arXiv:1610.09555, *Journal of Machine Learning Research (JMLR)* 20:1–6. (Cited on page 187).

Kovács, Ádám, Kinga Gémes, András Kornai, and Gábor Recski. 2022. "Explainable lexical entailment with semantic graphs." *Natural Language Engineering.* doi:https://www.doi.org/10.1017/S1351324922000092. (Cited on page 53).

Kovaleva, Olga, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. "Revealing the dark secrets of BERT." *arXiv preprint arXiv:1908.08593.* (Cited on page 120).

Krizhevsky, A., and G. Sutskever I.and Hinton. 2012. "ImageNet classification with deep convolutional neural networks." In *NIPS'2012.* (Cited on pages 115, 137).

Kuti, Judit, Enikő Héja, and Bálint Sass. 2010. "Sense disambiguation – "Ambiguous sensation"? Evaluating sense inventories for verbal WSD in Hungarian." In *Proceedings of LREC 2010 Workshop on Exploitation of Multilingual Resources and Tools for Central and (South-)Eastern European Languages,* edited by Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odjik, Stelios Piperidis, Mike Rosner, and Daniel Tapias, 23–30. European Language Resources Association (ELRA). (Cited on page 186).

Lahat, Dana, Tülay Adali, and Christian Jutten. 2015. "Multimodal data fusion: an overview of methods, challenges, and prospects." *Proceedings of the IEEE* 103 (9): 1449–1477. (Cited on page 181).

Landauer, T.K., P.W. Foltz, and D. Laham. 1998. "Introduction to Latent Semantic Analysis." *Discourse Processes* 25:259–284. (Cited on pages 80, 81).

Landauer, Thomas K, and Susan T Dumais. 1997. "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge." *Psychological review* 104 (2): 211. (Cited on page 176).

Langacker, Ronald. 1987. *Foundations of Cognitive Grammar.* Vol. 1. Stanford University Press. (Cited on page 20).

Lazaridou, Angeliki, Georgiana Dinu, and Marco Baroni. 2015. "Hubness and Pollution: Delving into Cross-Space Mapping for Zero-Shot Learning." In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers),* 270–280. Long, Oral. Beijing, China: Association for Computational Linguistics. doi:10.3115/v1/P15-1027. http://www.aclweb.org/anthology/P15-1027. (Cited on page 211).

Lazaridou, Angeliki, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2013. "Compositionally Derived Representations of Morphologically Complex Words in Distributional Semantics." In *ACL (1),* 1517–1526. http://aclweb.org/anthology/P/P13/P13-1149.pdf. (Cited on page 104).

Le, Q.V., and T. Mikolov. 2014. "Distributed Representations of Sentences and Documents." In *ICML.* (Cited on page 94).

Lebret, Rémi, and Ronan Collobert. 2015. "Rehabilitation of count-based models for word vector representations." In *International Conference on Intelligent Text Processing and Computational Linguistics,* 417–429. Springer. (Cited on page 108).

Lee, Lillian. 1999. "Distributional similarity models: Clustering vs. nearest neighbors." In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics,* 33–40. (Cited on page 88).

Lenat, Douglas B., and R.V. Guha. 1990. *Building Large Knowledge-Based Systems.* Addison-Wesley. (Cited on pages 22, 38, 65).

Lévai, Dániel, and András Kornai. 2019. "The impact of inflection on word vectors." In *XV. Magyar Számitógépes Nyelvészeti Konferencia.* January. (Cited on page 106).

Levelt, Willem J. M., Ardi Roelofs, and Antje S. Meyer. 1999. "A theory of lexical access in speech production." *Behavioral and brain sciences* 22:1–75. (Cited on pages v, 13, 64).

Levin, Beth. 1993. *English Verb Classes and Alternations: A Preliminary Investigation.* University of Chicago Press. (Cited on pages 31, 175, 195, 197).

Levy, Omer, and Yoav Goldberg. 2014a. "Dependency-Based Word Embeddings." In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 302–308. Baltimore, Maryland: Association for Computational Linguistics, June. http://www.aclweb.org/anthology/P14-2050. (Cited on page 176).

———. 2014b. "Linguistic Regularities in Sparse and Explicit Word Representations." In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning,* 171–180. Ann Arbor, Michigan: Association for Computational Linguistics. http://aclweb.org/anthology/W14-1618. (Cited on pages 94, 96, 137, 183).

———. 2014c. "Neural Word Embedding as Implicit Matrix Factorization." In *Advances in Neural Information Processing Systems 27,* edited by Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, 2177–2185. (Cited on pages 80, 94, 176).

Levy, Omer, Yoav Goldberg, and Ido Dagan. 2015. "Improving Distributional Similarity with Lessons Learned from Word Embeddings." *Transactions of the Association for Computational Linguistics* 3:211–225. doi:10.1162/tacl_a_00134. https://www.aclweb.org/anthology/Q15-1016. (Cited on pages 94, 95, 97).

Levy, Omer, Steffen Remus, Chris Biemann, and Ido Dagan. 2015. "Do Supervised Distributional Methods Really Learn Lexical Inference Relations?" In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 970–976. Denver, Colorado: Association for Computational Linguistics. doi:10.3115/v1/N15-1098. https://www.aclweb.org/anthology/N15-1098. (Cited on pages 94, 111, 178).

Li, Jiwei, and Dan Jurafsky. 2015. "Do Multi-Sense Embeddings Improve Natural Language Understanding?" In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing,* 1722–1732. Lisbon, Portugal: Association for Computational Linguistics, September. (Cited on pages 114, 183, 209, 210).

Linzen, Tal. 2016. "Issues in evaluating semantic spaces using word analogies." In *RepEval.* (Cited on page 112).

Liu, Hugo, and Push Singh. 2004. "ConceptNet—a practical commonsense reasoning tool-kit." *BT technology journal* 22 (4): 211–226. (Cited on page 46).

Ljubešić, Nikola, and Tomaž Erjavec. 2011. "hrWaC and slWac: Compiling Web Corpora for Croatian and Slovene." In *Text, Speech and Dialogue - 14th International Conference, TSD 2011, Pilsen, Czech Republic, September 1-5, 2011. Proceedings,* edited by Ivan Habernal and Václav Matousek, 395–402. Lecture Notes in Computer Science. Springer. (Cited on page 143).

Luong, Thang, Richard Socher, and Christopher D. Manning. 2013. "Better Word Representations with Recursive Neural Networks for Morphology." In *CoNLL,* 104–113. (Cited on page 104).

Maas, Andrew L, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. "Learning word vectors for sentiment analysis." In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1,* 142–150. Association for Computational Linguistics. (Cited on page 186).

MacAvaney, Sean, and Amir Zeldes. 2018. "A Deeper Look into Dependency-Based Word Embeddings." In *naacl.* (Cited on page 183).

Majewska, Olga, Ivan Vulić, Diana McCarthy, Yan Huang, Akira Murakami, Veronika Laippala, and Anna Korhonen. 2018. "Investigating the cross-lingual translatability of VerbNet-style classification." *Language Resources and Evaluation* 52 (3): 771–799. (Cited on pages 181, 197).

Makrai, Márton. 2013. "Fogalmak fontossága a definíciós gráf vizsgálatával [Importance of concepts based on the analysis of the definition graph." In *VII. Alkalmazott Nyelvészeti Doktoranduszkonferencia,* edited by Tamás Váradi. MTA Nyelvtudományi Intézet Budapest. ISBN: 978-963-9074-59-0. http://www.nytud.hu/alknyelvdok13/proceedings13/ANyD7-Makrai-Marton.pdf. (Cited on pages 61, 67).

———. 2014. "Deep cases in the `4lang` concept lexicon." In *X. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2014),* edited by Attila Tanács, Viktor Varga, and Veronika Vincze, 50–57 (in Hungarian), 387 (English abstract). ISBN: 978-963-306-246-3. (Cited on pages 91, 154, 167).

———. 2015. "Comparison of distributed language models on medium-resourced languages." In *XI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2015),* edited by Attila Tanács, Viktor Varga, and Veronika Vincze, 22–33. Szegedi Tudományegyetem Informatikai Tanszékcsoport. ISBN: 978-963-306-359-0. (Cited on pages 104, 107, 137, 138, 201).

———. 2016a. "Filtering Wiktionary triangles by linear mapping between distributed models." In Makrai 2016b. (Cited on pages 137, 186, 216).

Makrai, Márton. 2016b. “Filtering Wiktionary triangles by linear mapping between distributed models.” In *LREC.* (Cited on page 198).

———. 2020. “Tárgyas szerkezetek elemzése tenzorfelbontással– áttekintő cikk.” In *XVI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2020),* 273–287. Szeged. (Cited on page 177).

———. 2022. “Three-order normalized PMI and other lessons in tensor analysis of verbal selectional preferences.” In *XVIII. Magyar Számítógépes Nyelvészeti Konferencia,* edited by Gábor Berend, Gábor Gosztolya, and Veronika Vincze, 105–120. Szegedi Tudományegyetem TTIK, Informatikai Intézet. ISBN: 978-963-306-848-9. (Cited on page 194).

Makrai, Márton, and Veronika Lipp. 2018. “Do multi-sense word embeddings learn more senses?” In *K + K = 120 Workshop Dedicated to László Kálmán and András Kornai on the occasion of their 60th birthdays.* (Cited on pages 183, 198, 207, 216).

Makrai, Márton, Dávid Márk Nemeskey, and András Kornai. 2013. “Applicative structure in vector space models.” In *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality,* 59–63. Sofia, Bulgaria: ACL, August. http://www.aclweb.org/anthology/W13-3207. (Cited on page 148).

Manin, Dmitrii Y. 2008. “Zipf’s Law and Avoidance of Excessive Synonymy.” *Cognitive Science* 32 (7): 1075–1098. (Cited on page 177).

Marcus, Mitchell, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. “Building a Large Annotated Corpus of English: The Penn Treebank.” *Computational Linguistics* 19:313–330. (Cited on page 139).

McCann, Bryan, James Bradbury, Caiming Xiong, and Richard Socher. 2017. “Learned in translation: Contextualized word vectors.” In *Advances in Neural Information Processing Systems,* 6294–6305. (Cited on page 115).

McGill, William. 1954. “Multivariate information transmission.” *Transactions of the IRE Professional Group on Information Theory* 4 (4): 93–111. (Cited on page 179).

McInnes, Leland, John Healy, and Steve Astels. 2017. “hdbscan: Hierarchical density based clustering.” *The Journal of Open Source Software* 2, no. 11 (March). doi:10.21105/joss.00205. https://doi.org/10.21105%5C%2Fjoss.00205. (Cited on pages x, 195).

McInnes, Leland, John Healy, Nathaniel Saul, and Lukas Grossberger. 2018. “UMAP: Uniform Manifold Approximation and Projection.” *The Journal of Open Source Software* 3 (29): 861. (Cited on pages xii, 195).

Miháltz, Márton. 2005. "Towards A Hybrid Approach to Word-Sense Disambiguation in Machine Translation." In *RANLP-2005 Workshop: Modern Approaches in Translation Technologies,* edited by Galia Angelova, Kalina Bontcheva, Ruslan Mitkov, Nicolas Nicolov, and Nikolai Nikolov. September. ISBN: 954-91743-3-6. (Cited on page 186).

Miháltz, Márton, Csaba Hatvani, Judit Kuti, György Szarvas, János Csirik, Gábor Prószéky, and Tamás Váradi. 2008. "Methods and results of the Hungarian WordNet project." In *Proceedings of the Fourth Global WordNet Conference (GWC-2008).* Citeseer. (Cited on page 42).

Miháltz, Márton, and Bálint Sass. 2013. "What do we drink? Automatically extending Hungarian WordNet with selectional preference relations." In *Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora,* 105–109. (Cited on page 186).

Mikolov, Tomáš. 2010. *Recurrent neural network based language model.* Presentation at Google. Brno University of Technology. (Cited on page 92).

Mikolov, Tomas, Kai Chen, G.s. Corrado, and Jeffrey Dean. 2013. "Efficient Estimation of Word Representations in Vector Space." In *1st International Conference on Learning Representations, ICLR 2013, Workshop Track Proceedings,* edited by Y. Bengio and Y. LeCun. May. arXiv: 1301.3781 [cs.CL]. http://arxiv.org/abs/1301.3781. (Cited on pages 94, 95, 97, 104, 107, 137–139, 159, 178, 201, 212).

Mikolov, Tomas, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. "Advances in Pre-Training Distributed Word Representations." In *Language Resources and Evaluation Conference (LREC).* May. (Cited on page 110).

Mikolov, Tomas, Quoc V Le, and Ilya Sutskever. 2013. "Exploiting similarities among languages for machine translation." ArXiv preprint arXiv:1309.4168. (Cited on pages 94, 138, 142–144, 199, 200, 209, 210, 216).

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. "Distributed Representations of Words and Phrases and their Compositionality." In *Advances in Neural Information Processing Systems 26,* edited by C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, 3111–3119. Curran Associates, Inc. https://bit.ly/39HikH8. (Cited on pages 92–94, 137, 144, 176, 209).

Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig. 2013. "Linguistic Regularities in Continuous Space Word Representations." In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2013),* 746–751. Atlanta, Georgia: Association for Computational Linguistics. (Cited on pages 78, 92–94, 96, 103, 104, 111, 113, 137–139, 142, 212).

Miller, George A. 1995. "WordNet: a lexical database for English." *Communications of the ACM* 38 (11): 39–41. (Cited on pages 42, 61, 149, 154).

Mitchell, Jeff, and Mirella Lapata. 2008. "Vector-based Models of Semantic Composition." In *Proceedings of ACL-08: HLT,* 236–244. Columbus, Ohio: Association for Computational Linguistics. (Cited on page 186).

———. 2010. "Composition in Distributional Models of Semantics." *Cognitive Science* 34:1388–1429. (Cited on page 186).

Mnih, Andriy, and Geoffrey Hinton. 2007. "Three new graphical models for statistical language modelling." In *Proceedings of the 24th international conference on Machine learning,* 641–648. ACM. (Cited on pages 105, 178).

Mnih, Andriy, and Geoffrey E Hinton. 2008. "A scalable hierarchical distributed language model." *Advances in neural information processing systems* 21:1081–1088. (Cited on page 94).

———. 2009. "A scalable hierarchical distributed language model." *Advances in neural information processing systems* 21:1081–1088. (Cited on pages x, 150, 154).

Moens, Marc, and Mark Steedman. 1988. "Temporal Ontology and Temporal Reference." *Computational Linguistics* 14 (2): 15–28. https://www.aclweb.org/anthology/J88-2003. (Cited on page 56).

Mohammad, Saif, Bonnie Dorr, and Graeme Hirst. 2008. "Computing word-pair antonymy." In *Proceedings of the Conference on Empirical Methods in Natural Language Processing,* 982–991. Association for Computational Linguistics. (Cited on pages 101, 186).

Moravcsik, J. M. 1975. "Aitia as Generative Factor in Aristotle's Philosophy." *Dialogue* 14:622–36. (Cited on page 35).

Morin, Frederic, and Yoshua Bengio. 2005. "Hierarchical Probabilistic Neural Network Language Model." In *Aistats,* 5:246–252. Citeseer. (Cited on page 92).

Mrkšić, Nikola, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. "Counter-fitting Word Vectors to Linguistic Constraints." In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 142–148. San Diego, California: Association for Computational Linguistics, June. (Cited on page 101).

Mrkšić, Nikola, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017. "Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints." *Transactions of the Association for Computational Linguistics* 5:309–324. (Cited on pages 101, 102).

Nakov, Preslav, Antonia Popova, and Plamen Mateev. 2001. "Weight functions impact on LSA performance." *EuroConference RANLP:* 187–193. (Cited on page 80).

Neelakantan, Arvind, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. "Efficient Non-parametric Estimation of Multiple Embeddings per Word in Vector Space." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* 1059–1069. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/D14-1113. http://www.aclweb.org/anthology/D14-1113. (Cited on pages 183, 209).

Nemeskey, Dávid Márk. 2017. "emLam – a Hungarian Language Modeling baseline." In *XIII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2017),* 91–102. Szeged. arXiv: 1701.07880 [cs.CL]. (Cited on pages 106, 216).

Nemeskey, Dávid, Gábor Recski, Márton Makrai, Attila Zséder, and András Kornai. 2013. "Spreading activation in language understanding." In *Proceedings of the 9th International Conference on Computer Science and Information Technologies (CSIT 2013),* 140–143. Yerevan, Armenia: Springer. https://hlt.bme.hu/media/pdf/nemeskey_2013.pdf. (Cited on page 7).

Ng, Andrew Y., Michael I. Jordan, and Yair Weiss. 2001. "On Spectral Clustering: Analysis and an algorithm." In *Advances in neural information processing systems,* 849–856. MIT Press. (Cited on page 152).

Nivre, Joakim, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, et al. 2016. "Universal Dependencies v1: A Multilingual Treebank Collection." In *Proc. LREC 2016,* 1659–1666. May. (Cited on page 187).

Niwa, Yoshiki, and Yoshihiko Nitta. 1994. "Cooccurrence vectors from corpora vs. distance vectors from dictionaries." In *Proceedings of the 15th conference on Computational linguistics-Volume 1,* 304–309. Association for Computational Linguistics. (Cited on page 79).

Hendrix, G. G. 1975. *Partitioned Networks for the Mathematical Modeling of Natural Language Semantics.* Technical report. Department of Computer Science, University of Texas at Austin. (Cited on page 64).

Ogden, C.K. 1944. *Basic English: A General Introduction with Rules and Grammar.* Psyche miniatures: General Series. Kegan Paul, Trench, Trubner. http://books.google.hu/books?id=l-EtAAAAYAAJ. (Cited on pages 65, 66).

Olshausen, Bruno A, and David J Field. 1997. "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision research* 37 (23): 3311–3325. (Cited on page 157).

Oravecz, Csaba, Tamás Váradi, and Bálint Sass. 2014. "The Hungarian Gigaword Corpus." In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014).* Reykjavik, Iceland: European Language Resources Association (ELRA). http://www.aclweb.org/anthology/L14-1536. (Cited on pages x, 143, 147, 201, 212).

Osgood, Charles E., William S. May, and Murray S. Miron. 1975. *Cross Cultural Universals of Affective Meaning.* University of Illinois Press. (Cited on page 79).

Ostler, Nicholas. 1979. *Case-Linking: a Theory of Case and Verb Diathesis Applied to Classical Sanskrit.* MIT: PhD thesis. (Cited on page 152).

Palmer, Martha, Daniel Gildea, and Nianwen Xue. 2010. "Semantic role labeling." *Synthesis Lectures on Human Language Technologies* 3 (1): 1–103. (Cited on pages 21, 27, 43).

Panchenko, A, E Ruppert, S Faralli, S.P Ponzetto, and C Biemann. 2018. "Building a Web-Scale Dependency-Parsed Corpus from Common Crawl." In *Proceedings of LREC 2018.* ELRA. (Cited on page 187).

Panchenko, Alexander, Johannes Simon, Martin Riedl, and Chris Biemann. 2016. "Noun Sense Induction and Disambiguation using Graph-Based Distributional Semantics." In *KONVENS Conference on NLP, Germany.* October. (Cited on page 181).

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, et al. 2011. "Scikit-learn: Machine Learning in Python." *Journal of Machine Learning Research* 12:2825–2830. (Cited on pages 161, 196).

Pennington, Jeffrey, Richard Socher, and Christopher Manning. 2014. "Glove: Global Vectors for Word Representation." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* 1532–1543. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/D14-1162. http://www.aclweb.org/anthology/D14-1162. (Cited on pages 95, 96, 176, 178, 212).

Perlmutter, David M. 1983. *Studies in Relational Grammar.* University of Chicago Press. (Cited on page 152).

Peters, Matthew, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. "Deep Contextualized Word Representations." In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers),* 2227–2237. New Orleans, Louisiana: Association for Computational Linguistics. doi:10.18653/v1/N18-1202. http://aclweb.org/anthology/N18-1202. (Cited on pages 105, 115, 128, 137, 183).

Peters, Matthew, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018. "Dissecting Contextual Word Embeddings: Architecture and Representation." In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,* 1499–1509. Brussels, Belgium: Association for Computational Linguistics. http://aclweb.org/anthology/D18-1179. (Cited on page 183).

Pilehvar, Mohammad Taher, and Nigel Collier. 2016. "De-Conflated Semantic Representations." Http://arxiv.org/abs/1608.01961v1, August 5. arXiv: http://arxiv.org/abs/1608.01961v1 [cs.CL, cs.AI]. http://arxiv.org/abs/1608.01961v1. (Cited on page 181).

Polajnar, Tamara, Laura Rimell, and Stephen Clark. 2014. "Using Sentence Plausibility to Learn the Semantics of Transitive Verbs." In *NIPS Learning Semantics Workshop.* In arXiv, some minor errata fixed. (Cited on page 176).

Pustejovsky, James. 1995. *The Generative Lexicon.* MIT Press. (Cited on pages 5, 33).

Pusztai, Ferenc, ed. 2003. *Magyar értelmező kéziszótár.* Akadémiai Kiadó. (Cited on page 209).

Putnam, H. 1976. "Two Dogmas Revisited." *Printed in his (1983) Realism and Reason, Philosophical Papers* 3. (Cited on page 72).

Qi, Peng, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. "Stanza: A Python Natural Language Processing Toolkit for Many Human Languages." arXiv: 2003.07082. https://arxiv.org/abs/2003.07082. (Cited on page 53).

Qiu, Xipeng, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. "Pre-trained models for natural language processing: A survey." *arXiv preprint arXiv:2003.08271.* (Cited on page 115).

Quillian, M. Ross. 1968. "Word concepts: A theory and simulation of some basic semantic capabilities." *Behavioral Science* 12:410–430. (Cited on page 6).

———. 1969. "The teachable language comprehender." *Communications of the ACM* 12:459–476. (Cited on pages xii, 6, 60).

Quine, W.V. 1969. "Natural kinds." In *In Ontological Relativity and other essays.* Columbia University Press. (Cited on page 39).

Quine, Willard van Orman. 1951. "Two dogmas of empiricism." *The Philosophical Review* 60:20–43. (Cited on page 71).

Rabanser, Stephan, Oleksandr Shchur, and Stephan Günnemann. 2017. "Introduction to Tensor Decompositions and their Applications in Machine Learning." ArXiv:1711.10781 [stat.ML], November. arXiv: http://arxiv.org/abs/1711.10781v1 [stat.ML, cs.LG]. http://arxiv.org/abs/1711.10781v1. (Cited on page 180).

Radford, Alec, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. "Improving language understanding by generative pre-training." https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf. (Cited on page 116).

Radovanović, M, A Nanopoulos, and M Ivanović. 2010. "Hubs in space: Popular nearest neighbors in high-dimensional data." *Journal of Machine Learning Research* 11:2487–2531. (Cited on page 210).

Ramachandran, Prajit, Peter J Liu, and Quoc V Le. 2017. "Unsupervised pretraining for sequence to sequence learning." In *EMNLP.* (Cited on page 115).

Recski, Gábor, Eszter Iklódi, Katalin Pajkossy, and András Kornai. 2016. "Measuring Semantic Similarity of Words Using Concept Networks." In *Proceedings of the 1st Workshop on Representation Learning for NLP,* 193–200. Berlin, Germany: Association for Computational Linguistics. doi:10.18653/v1/W16-1622. https://www.aclweb.org/anthology/W16-1622. (Cited on pages 60, 75).

Řehůřek, Radim, and Petr Sojka. 2010. "Software Framework for Topic Modelling with Large Corpora" [in English]. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks,* 45–50. Valletta, Malta: ELRA, May. http://is.muni.cz/publication/884893/en. (Cited on page 110).

Reisinger, Joseph, and Raymond J Mooney. 2010. "Multi-prototype vector-space models of word meaning." In *The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics,* 109–117. Association for Computational Linguistics. (Cited on pages 183, 209).

Al-Rfou', Rami, Bryan Perozzi, and Steven Skiena. 2013. "Polyglot: Distributed Word Representations for Multilingual NLP." In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning,* 183–192. Sofia, Bulgaria: Association for Computational Linguistics, August. http://www.aclweb.org/anthology/W13-3520. (Cited on page 155).

Rogers, Anna, Aleksandr Drozd, and Bofang Li. 2017. "The (too many) problems of analogical reasoning with word vectors." In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (* SEM 2017),* 135–148. (Cited on pages 112, 113).

Rogers, Anna, Olga Kovaleva, and Anna Rumshisky. 2020. "A primer in bertology: What we know about how bert works." *arXiv preprint arXiv:2002.12327.* (Cited on pages 116, 123, 126, 197).

Rubinstein, Ron, Michael Zibulevsky, and Michael Elad. 2008. "Efficient implementation of the K-SVD algorithm and the Batch-OMP method." *Department of Computer Science, Technion, Israel, Tech. Rep.* (Cited on page 157).

Ruder, Sebastian. 2018. *NLP's ImageNet moment has arrived.* https://ruder.io/nlp-imagenet/. (Cited on pages 105, 115).

Ruhl, C. 1989. *On monosemy: a study in lingusitic semantics.* State University of New York Press. (Cited on page 62).

Rumelhart, D., and J. McClelland. 1986. "On learning the past tenses of English verbs." In *Parallel distributed processing: Explorations in the microstructure of cognition,* edited by D. Rumelhart and J. McClelland. Cambridge MA: Branford. (Cited on page 89).

Russell, Stuart, and Peter Norvig. 2002. "Artificial intelligence: a modern approach." (Cited on page 42).

Rychlý, Pavel. 2008. "A Lexicographer-Friendly Association Score." In *Proceedings of Recent Advances in Slavonic Natural Language Processing,* 6–9. (Cited on page 178).

Sahlgren, M. 2006. "The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces." PhD diss., Department of Linguistics, Stockholm University. (Cited on pages 78, 81–84, 97).

Salle, Alexandre, and Aline Villavicencio. 2018. "Incorporating Subword Information into Matrix Factorization Word Embeddings." *arXiv preprint arXiv:1805.03710.* (Cited on page 105).

Salton, Gerard, Anita Wong, and Chung-Shu Yang. 1975. "A vector space model for automatic indexing." *Communications of the ACM* 18 (11): 613–620. (Cited on page 79).

Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter." arXiv: 1910.01108 [cs.CL]. (Cited on page 87).

Saralegi, Xabier, Iker Manterola, and Iñaki San Vicente. 2011. "Analyzing methods for improving precision of pivot based bilingual dictionaries." In *Proceedings of the Conference on Empirical Methods in Natural Language Processing,* 846–856. Association for Computational Linguistics. (Cited on page 200).

Sass, Bálint. 2015. "28 millió szintaktikailag elemzett mondat és 500000 igei szerkezet [28 million syntactically analyzed sentences and 500 000 verb constructions in Hungarian]." In *XI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2015),* edited by Tanács Attila, Varga Viktor, and Vincze Veronika, 303–308. Szegedi Tudományegyetem Informatikai Tanszékcsoport. ISBN: 978-963-306-359-0. (Cited on pages 175, 186, 196).

Sass, Bálint. 2018. "A Lattice Based Algebraic Model for Verb Centered Constructions." In *TSD,* 231–238. Springer. (Cited on page 186).

Schank, Roger C. 1972. "Conceptual dependency: A theory of natural language understanding." *Cognitive Psychology* 3 (4): 552–631. (Cited on pages 13, 66).

———. 1973. *The Fourteen Primitive Actions and Their Inferences.* Stanford AI Lab Memo 183. (Cited on pages 13, 14).

———. 1975. *Conceptual Information Processing.* North-Holland. (Cited on pages 18, 60).

Schnabel, Tobias, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. "Evaluation methods for unsupervised word embeddings." In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing,* 298–307. (Cited on pages 114, 186).

Schuster, Sebastian, and Christopher D. Manning. 2016. "Enhanced English Universal Dependencies: An Improved Representation for Natural Language Understanding Tasks." In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16),* 2371–2378. Portorož, Slovenia: European Language Resources Association (ELRA). https://www.aclweb.org/anthology/L16-1376. (Cited on page 53).

Schütze, Hinrich. 1993. "Word Space." In *Advances in Neural Information Processing Systems 5,* edited by SJ Hanson, JD Cowan, and CL Giles, 895–902. Morgan Kaufmann. (Cited on page 83).

———. 1998. "Automatic Word Sense Discrimination." *Computational Linguistics Special-Issue-on-Word Sense Disambiguation* 24 (1). http://www.aclweb.org/anthology/J98-1004. (Cited on page 207).

Sellars, Roy Wood. 1917. *The essentials of logic.* Houghton Mifflin. (Cited on page 37).

Sen, M.U., and H. Erdogan. 2014. "Learning word representations for Turkish." In *Signal Processing and Communications Applications Conference (SIU), 2014 22nd,* 1742–1745. April. doi:10.1109/SIU.2014.6830586. (Cited on page 138).

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016. "Neural Machine Translation of Rare Words with Subword Units." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* 1715–1725. Berlin, Germany: Association for Computational Linguistics, August. doi:10.18653/v1/P16-1162. https://www.aclweb.org/anthology/P16-1162. (Cited on page 105).

Shannon, Claude E., and Warren W. Weaver. 1949. *The Mathematical Theory of Communication.* Urbana: University of Illinois Press. (Cited on page 179).

Sharan, Vatsal, and Gregory Valiant. 2017. "Orthogonalized ALS: A Theoretically Principled Tensor Decomposition Algorithm for Practical Use." In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017,* 3095–3104. August. http://proceedings.mlr.press/v70/sharan17a.html. (Cited on pages 176, 178, 180, 182, 184, 196).

Sidiropoulos, Nicholas D., Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E. Papalexakis, and Christos Faloutsos. 2017. "Tensor Decomposition for Signal Processing and Machine Learning." *IEEE Transactions on signal processing* (Piscataway, NJ, USA) 65, no. 13 (July): 3551–3582. ISSN: 1053-587X. doi:10.1109/TSP.2017.2690524. https://doi.org/10.1109/TSP.2017.2690524. (Cited on page 180).

Siklósi, Borbála. 2016. “Using embedding models for lexical categorization in morphologically rich languages.” In *International Conference on Intelligent Text Processing and Computational Linguistics,* 115–126. Springer. (Cited on page 186).

Sinclair, John M. 1987. *Looking up: an account of the COBUILD project in lexical computing.* Collins ELT. (Cited on pages ix, 208).

Smith, J. Maynard. 1974. “Theory of games and the evolution of animal conflicts.” *Journal of Theoretical Biology* 47:209–221. (Cited on pages 11, 12).

Smith, Noah A. 2019. “Contextual Word Representations: A Contextual Introduction.” ArXiv:1902.06006. (Cited on page 105).

Smolensky, Paul. 1990. “Tensor product variable binding and the representation of symbolic structures in connectionist systems.” *Artificial intelligence* 46 (1): 159–216. (Cited on page 89).

Socher, R., M. Ganjoo, H. Sridhar, O. Bastani, C. D. Manning, and A. Y. Ng. 2013. “Zero-shot learning through cross-modal transfer.” In *International Conference on Learning Representations (ICLR 2013).* (Cited on page 101).

Somers, Harold L. 1987. *Valency and case in computational linguistics.* Edinburgh University Press. (Cited on pages 168, 170, 171).

Soricut, Radu, and Franz Och. 2015. “Unsupervised morphology induction using word embeddings.” In *Proceedings of NAACL,* 1627–1637. Denver, Colorado. (Cited on page 104).

Sowa, JF. 1976. “Conceptual Graphs for a Data Base Interface.” *Journal of Research and Development.* (Cited on page 15).

Sowa, John F. 1992. “Conceptual graphs as a universal knowledge representation.” *Computers & Mathematics with Applications* 23 (2): 75–93. (Cited on pages 15, 16).

Speer, Robert, Joshua Chin, and Catherine Havasi. 2017. “ConceptNet 5.5: An Open Multilingual Graph of General Knowledge.” In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17),* 4444–4451. (Cited on page 46).

Speer, Robert, and Catherine Havasi. 2012. “Representing General Relational Knowledge in ConceptNet 5.” In *LREC,* 3679–3686. (Cited on pages 46, 47).

Subramanian, Anant, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2018. “SPINE: SParse Interpretable Neural Embeddings.” *AAAI.* (Cited on page 157).

Sun, Lin, Anna Korhonen, Thierry Poibeau, and Cédric Messiant. 2010. "Investigating the cross-linguistic potential of VerbNet: style classification." In *Proceedings of the 23rd International Conference on Computational Linguistics,* 1056–1064. Association for Computational Linguistics. (Cited on pages 181, 197).

Suzuki, Ikumi, Kazuo Hara, Masashi Shimbo, Marco Saerens, and Kenji Fukumizu. 2013. "Centering Similarity Measures to Reduce Hubs." In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing,* 613–623. Seattle, Washington, USA: Association for Computational Linguistics. http://www.aclweb.org/anthology/D13-1058. (Cited on page 210).

Swadesh, Morris. 1950. "Salish internal relationships." *International Journal of American Linguistics:* 157–167. (Cited on page 65).

Szántó, Zsolt, Veronikag Vincze, and Richárd Farkas. 2017. "Magyar nyelvű szó- és karakterszintű szóbeágyazások [Word- and character-level word embeddings for Hungarian]." In *XIII. Magyar Számítógépes Nyelvészeti Konferencia,* 323–328. January. (Cited on page 186).

Szécsényi, Tibor. 2019. "Argumentumszerkezet-variánsok korpusz alapú meghatározása [Corpus-based identification of Hungarian argument structure variants]." In *XV. Magyar Számítógépes Nyelvészeti Konferencia,* edited by Gábor Berend, Gábor Gosztolya, and Veronika Vincze, 315–331. Szegedi Tudományegyetem TTIK, Informatikai Intézet, January. (Cited on page 196).

Takala, Pyry. 2016. "Word embeddings for morphologically rich languages." In *Proceedings of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.* (Cited on page 103).

Talmy, L. 1988. "Force dynamics in language and cognition." *Cognitive science* 12 (1): 49–100. (Cited on pages 20, 25).

Tanaka, Kumiko, and Kyoji Umemura. 1994. "Construction of a bilingual dictionary intermediated by a third language." In *Proceedings of the 15th conference on Computational linguistics-Volume 1,* 297–303. Association for Computational Linguistics. (Cited on pages 199, 200).

Tang, Gongbo, Rico Sennrich, and Joakim Nivre. 2019. "Encoders Help You Disambiguate Word Senses in Neural Machine Translation." In *EMNLP.* (Cited on page 129).

Tesniére, Lucien. 1959. *Élements de syntaxe structurale.* Paris: Klincksieck. (Cited on page 7).

Thorndike, Edward L. 1921. *The teacher's word book.* New York Teachers College, Columbia University. (Cited on page 65).

Tiedemann, Jörg. 2012. "Parallel Data, Tools and Interfaces in OPUS." In *LREC,* edited by Nicoletta Calzolari. Istanbul, Turkey: European Language Resources Association (ELRA), May. ISBN: 978-2-9517408-7-7. http://www.lrec-conf.org/proceedings/lrec2012/summaries/463.html. (Cited on pages 143, 200).

Tomašev N., Mladenić D. 2013. "Hub Co-occurrence Modeling for Robust High-Dimensional *k*NN Classification." In *Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2013. Lecture Notes in Computer Science,* edited by Blockeel H., Kersting K., Nijssen S., and Železný F., 8189:643–659. Springer, Berlin, Heidelberg. (Cited on page 210).

Trier, J. 1931. *Der deutsche Wortschatz im Sinnbezirk des Verstandes: die Geschichte eines sprachlichen Feldes. Band I: Von den Anfägen biz zum Beginn des 13. Jahrhunderts.* C. Winter. (Cited on page 36).

Tsvetkov, Yulia, Manaal Faruqui, and Chris Dyer. 2016. "Correlation-based intrinsic evaluation of word vector representations." *arXiv preprint arXiv:1606.06710.* (Cited on page 80).

Tucker, Ledyard R. 1966. "Some mathematical notes on three-mode factor analysis." *Psychometrika* 31 (3): 279–311. (Cited on page 181).

Turian, Joseph, Lev-Arie Ratinov, and Yoshua Bengio. 2010. "Word Representations: A Simple and General Method for Semi-Supervised Learning." In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics,* 384–394. Uppsala, Sweden: Association for Computational Linguistics. (Cited on pages 114, 154).

Turney, Peter D. 2006. "Similarity of semantic relations." *Computational Linguistics* 32:379–416. (Cited on page 138).

Turney, Peter D., and Patrick Pantel. 2010. "From Frequency to Meaning: Vector Space Models of Semantics." *Journal of Artificial Intelligence Research* 37:141–188. (Cited on pages 78, 79, 84, 86, 88, 176, 178).

Van de Cruys, Tim. 2009. "A Non-negative Tensor Factorization Model for Selectional Preference Induction." In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics,* 83–90. Athens, Greece: Association for Computational Linguistics, March. https://www.aclweb.org/anthology/W09-0211. (Cited on pages 175, 176, 178, 179, 183, 185).

———. 2011. "Two Multivariate Generalizations of Pointwise Mutual Information." In *Proceedings of the Workshop on Distributional Semantics and Compositionality,* 16–20. Portland, Oregon, USA: Association for Computational Linguistics, June. https://www.aclweb.org/anthology/W11-1303. (Cited on pages 178, 179, 182).

Van de Cruys, Tim, Thierry Poibeau, and Anna Korhonen. 2013. "A Tensor-based Factorization Model of Semantic Compositionality." In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 1142–1151. Atlanta, Georgia: Association for Computational Linguistics, June. https://www.aclweb.org/anthology/N13-1134. (Cited on pages 176, 178, 179, 182, 183, 185).

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention is All you Need." In *Advances in Neural Information Processing Systems 30,* edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, 5998–6008. Curran Associates, Inc. arXiv: 1706.03762 [cs.CL]. http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf. (Cited on page 116).

Vendler, Zeno. 1967. *Linguistics and Philosophy.* Ithaca, NY: Cornell University Press. (Cited on page 34).

Villada Moirón, M. B. 2005. "Data-driven identification of fixed expressions and their modifiability." PhD diss., University of Groningen. (Cited on page 179).

Vincze, Veronika, György Szarvas, Attila Almási, Dóra Szauter, Róbert Ormándi, Richárd Farkas, Csaba Hatvani, and János Csirik. 2008. "Hungarian Word-Sense Disambiguated Corpus." In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08),* edited by N Calzolari, ChoukriK, B Maegaard, J Mariani, J Odjik, S Piperidis, and D Tapias, 3344–3349. European Language Resources Association (ELRA). (Cited on page 186).

Virpioja, Sami, Peter Smit, Stig-Arne Grönroos, Mikko Kurimo, et al. 2013. "Morfessor 2.0: Python implementation and extensions for Morfessor Baseline." (Cited on page 106).

Vossen, P., W. Meijs, and M. den Broeder. 1989. "Meaning and structure in dictionary definitions." In *Computational lexicography for natural language processing,* 171–192. (Cited on page 41).

Vulić, Ivan, Nikola Mrkšić, and Anna Korhonen. 2017. "Cross-lingual induction and transfer of verb classes based on word vector space specialisation," *arXiv preprint arXiv:1707.06945* (Copenhagen, Denmark) (September): 2546–2558. doi:10.18653/v1/D17-1270. https://www.aclweb.org/anthology/D17-1270. (Cited on pages 181, 197).

Wang, Bin, and C.-C. Jay Kuo. 2020. *SBERT-WK: A Sentence Embedding Method by Dissecting BERT-based Word Models.* arXiv: 2002.06652 [cs.CL]. (Cited on page 133).

Watanabe, Satosi. 1960. "Information theoretical analysis of multivariate correlation." *IBM Journal of research and development* 4 (1): 66–82. (Cited on page 179).

Weinreich, U. 1964. "Webster's Third: A Critique of its Semantics." *International Journal of American Linguistics* 30:405–409. (Cited on page 33).

Whitney, William Dwight. 1885. "The roots of the Sanskrit language." *Transactions of the American Philological Association (1869–1896)* 16:5–29. (Cited on page 66).

Wierzbicka, Anna. 1972. *Semantic Primitives.* Frankfurt: Athenäum. (Cited on pages 22, 60).

———. 1985. *Lexicography and conceptual analysis.* Ann Arbor: Karoma. (Cited on pages 19, 66).

Wilks, Y., D. Fass, C. Guo, J. McDonald, T. Plate, and B. Slator. 1989. "A tractable machine dictionary as a resource for computational semantics," 193–228. (Cited on page 41).

Wilks, Yorick. 1977. "What sort of taxonomy of causation do we need for language understanding?" *Cognitive Science* 1 (3): 235–264. (Cited on page 18).

Winograd, T. 1972. "Understanding natural language." *Cognitive Psychology* 3 (1): 1–191. ISSN: 0010-0285. (Cited on page 6).

Woods, William A. 1975. "What's in a link: Foundations for semantic networks." *Representation and Understanding: Studies in Cognitive Science:* 35–82. (Cited on pages 15, 64, 75).

Xing, Chao, Chao Liu, Dong Wang, and Yiye Lin. 2015. "Normalized Word Embedding and Orthogonal Transform for Bilingual Word Translation." In *NAACL,* 1005–1010. (Cited on pages 201, 211, 214).

Yogatama, Dani, Manaal Faruqui, Chris Dyer, and Noah A. Smith. 2015. "Learning Word Representations with Hierarchical Sparse Coding." In *ICML.* Previous version in NIPS Deep Learning and Representation Learning Workshop 2014. (Cited on page 158).

Youn, Hyejin, Logan Sutton, Eric Smith, Cristopher Moore, Jon F. Wilkins, Ian Maddieson, William Croft, and Tanmoy Bhattacharya. 2016. "On the universal structure of human lexical semantics." *PNAS* 113 (7): 1766–1771. (Cited on pages 24, 208).

Zellers, Rowan, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. "HellaSwag: Can a Machine Really Finish Your Sentence?" *arXiv preprint arXiv:1905.07830.* (Cited on page 46).

Zgusta, Ladislav. 1971. *Manual of lexicography.* Prague: Academia. (Cited on page 208).

Zhang, Jingwei, Jeremy Salwen, Michael Glass, and Alfio Gliozzo. 2014. "Word Semantic Representations using Bayesian Probabilistic Tensor Factorization." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* 1522–1531. Doha, Qatar: Association for Computational Linguistics, October. doi:10.3115/v1/D14-1161. https://www.aclweb.org/anthology/D14-1161. (Cited on pages 184, 186).

Zhao, Peng, Gulherme Rocha, and Bin Yu. 2009. "The composite and absolute penalties for grouped and hierarchical variable selection." *The Annals of Statistics* 37(6A):3468–3497. (Cited on page 158).

Zhu, Yi, Ivan Vulić, and Anna Korhonen. 2019. "A Systematic Study of Leveraging Subword Information for Learning Word Representations." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers),* 912–932. Minneapolis, Minnesota: Association for Computational Linguistics, June. doi:10.18653/v1/N19-1097. https://www.aclweb.org/anthology/N19-1097. (Cited on page 104).

Zhuang, Yimeng, Jinghui Xie, Yinhe Zheng, and Xuan Zhu. 2018. "Quantifying Context Overlap for Training Word Embeddings." In *EMNLP.* (Cited on pages 178, 184).

Zséder, Attila, Gábor Recski, Dániel Varga, and András Kornai. 2012. "Rapid creation of large-scale corpora and frequency dictionaries." In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12),* 1462–1465. Istanbul, Turkey: European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2012/pdf/783_Paper.pdf. (Cited on pages 143, 144).

Zsibrita, János, Veronika Vincze, and Richárd Farkas. 2013. "magyarlanc: A Tool for Morphological and Dependency Parsing of Hungarian." In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP 2013),* 763–771. Hissar, Bulgaria: INCOMA Ltd. Shoumen. (Cited on page 106).