Contents

Todo list

Chapter 1

Introduction

Computers, machines that play a more and more important role in our lives, require specially designed programming languages to be controlled. This is different from interactions between people where spoken or written natural language is used. Ideally, it would be perfect if interactions with computers was not different to interaction between humans. Computational linguistics is one of the fields that aims to solve this problem.

In order to be controlled by people, or be able to assist people in language related tasks, computers need to understand it. However, different tasks need various level of language "understanding". For instance, even if one does not recognize or know the language of a piece of text on Figure 1.1a, he or she can tell how many words there are, and that there is only one sentence. After a while, one can even say that this is probably a piece of poetry.

The conclusions above require neither deep understanding of the language nor the meaning of the text. Knowledge that texts (at least in some languages) consist of words separated by a space and how poems are usually written is enough. Moreover, knowing the letter combination distribution across the languages or a list of words for all human languages, one would conclude that the text on Figure 1.1a is in Latvian. We managed to get this answers without knowing what the text is about.

On the other hand, a task that provides a list of associations with the text, an essay or a painting inspired by it demands understanding of the text, language knowledge. Luckily, nowadays these kind of problems are not expected or demanded to be done by computers. People enjoy doing these kind of tasks themselves.

However, it is reasonable to ask a computer the following questions regarding text meaning: a) What is the text on Figure 1.1a about? b) What is the relation between the texts on Figure 1.1a and Figure 1.1b? c) Are these texts identical? d) Where did the meeting took place? e) What poems are similar to this?

Text summarizaiton, machine translation, information extraction and retrieval are just a few of many branches of computational linguistics that provide methods for answering these questions. The questions above have a general property: all of them are about the

Jaunkundze ar sunīti

Un Vecrīgas šķērsielā, šaurā
kā vēstuļu kastītes sprauga,
kur troksnim un burzmai tik atbalss,
kur smaržo pēc darvas,
dzelzs un pēc āboliem pagrabos sausos,
es satiku jaunkundzi –
glītu un veiklu kā mēle,
kā spēlējot vijoles lociņš.

(a)

Барышня с собачкой

В Старой Риге, на улице поперечной, узкой,
как щель в почтовый ящик,
в который проникают только отголоски шума, гама,
где запах дёгтя, ржавчины и яблок в сухих подвалах,
я встретил барышню –
красива и ловка - она - язык,
смычок, играющий на скрипке.

(b)

Young Woman with a Dog

On a narrow side-street in Riga's old quarter,
as though in a mailbox slot
where noise and hustle only echo,
and it smells of tar and steel
and apples kept in dry basements,

I met a young woman
attractive and active
as a tongue,
as a violin-bow playing.

(c)

Figure 1.1: Three pieces of written natural language. The text on Figure 1.1a is the beginning of the poem "Jaunkundze ar sunīti" by Aleksandrs Čaks, Figure 1.1b is a translation to Russian by Lora Trin, and Figure 1.1c is an English translation by Inara Cedrins.

meaning of the text. Natural language semantics is an area that studies meaning representation.

Creativity of natural languages—the fact that humans are able to produce and understand sentences they have never came across—complicates meaning modeling. Even if we had a way to map each word to its meaning, it is impractical to apply the same procedure to sentences, because as we process a piece of text most of the sentences in it will be seen for the fist time. Because of this, we need to be able to build the meaning representation of a sentence from its constituent parts: words.

Syntax is a study about the structure of a sentence. Grammars define the rules that are describe how a sentences that belong to a language should look like. For example, a subject is in front of a verb and an object is after it in an English sentence. Having the constituent meaning representation, the meaning of a sentence is built guided by its syntax.

To a first, high level approximation, to be able to deal with the meaning of a text in natural language one needs to have meaning representation of constituents, a view to the (syntactic) structure of the text and a compositional procedure that outputs the meaning representation of the whole text.

Finish introduction

Chapter 2

Overview

2.1 Similarity

Similarity is the degree of resemblance between two objects or events Hahn (2014) and plays a crucial role in psychological theories of knowledge and behaviour, where it is used to explain such phenomena as classification and conceptualisation. *Fruit* is a *category* because it is a practical generalisation. Fruits are sweet and constitute deserts, so when one is presented with an unseen fruit, one can hypothesise that it is served towards the end of a dinner.

Generalisations are extremely powerful in describing a language as well. The verb *runs* requires its subject to be singular. *Verb*, *subject* and *singular* are categories that are used to describe English grammar. When one encounters an unknown word and is told that it is a verb, one will immediately have an idea how to use it assuming that it is used similarly to other English verbs.

Semantic formalisation of similarity bases on two ideas. Word occurrence patterns *define* its meaning Firth (1957), while the difference in occurrence quantifies the difference in meaning Harris (1970). From computational perspective, this motivates and guides development of similarity components that are embedded into natural language processing systems that deal with tasks such as word sense disambiguation Schütze (1998), information retrieval Salton et al. (1975) and machine translation Dagan et al. (1993).

Because it is difficult to measure performance of a single (similarity) component in a pipeline, datasets that focus on similarity are popular among computational linguists. Apart from a pragmatic attempt to alleviate the evaluation of similarity components, these datasets serve as an empirical test of the hypotheses of Firth and Harris, merging together our understanding of human mind, language and technology.

2.2 Formal semantics

Formal approaches to the semantics of natural language have long built upon the classical idea of compositionality – that the meaning of a sentence is a function of its parts **?**. In compositional type-logical approaches, predicate-argument structures representing phrases and sentences are built from their constituent parts by general operations such as beta-reduction within the lambda calculus **?**: for example, given a semantic representation of *"John"* as $john'$ and *"loves"* as $\lambda y.\lambda x.loves'(x, y)$, the sentence *"John loves Mary"* can be constructed as $\lambda y.\lambda x.loves'(x, y)(mary')(john') = loves'(john', mary')$.

To get the semantic representation of the sentence *John loves Mary* we need to do the following. Syntactic rules define how constituents are combined to form other constituents (and finally a sentence). Translation rules define how semantic representations of the constituents are combined to get a semantic representation of the whole.

In semantics of natural language, categorial grammars are widely used to obtain syntactic structure of a sentence. Given a set of basic categories `ATOM`, for example $\{n, s, np\}$ complex categories `CAT`$\backslash$`CAT` and `CAT`/`CAT` can be constructed, where `CAT` is either an element of `ATOM` or a complex category. So the transitive verb category is `np`$\backslash$`s`/`np`. Intuitively we want to say that obtaining a sentence with a transitive verb there must be a noun phrase before and after it.

Parsing is done by composing categories together according to two rules:

1. **Backward application**: If α is a string of category A and β is a string of category $A\backslash B$, then $\alpha\beta$ is of category B.
2. **Forward application**: If α is a string of category A and β is a string of category B/A, then $\beta\alpha$ is of category B.

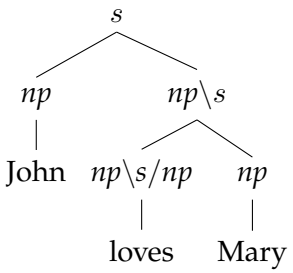

Figure 2.1: A syntactic tree for *John loves Mary*. Lexicon assigns categories to words: *John* is *np*, loves is *np*$\backslash$*s*/*np* and Mary is *np*. Backward and forward composition rules derive the syntactic tree.

Figure 2.1 illustrates the parse tree for *John loves Mary* obtained using the category composition rules.

The last step is to map syntactic categories with semantic terms. Again, there are base types (e for entities and t for sentences) and complex types of the form $(a \to b)$ where a and b are types. The mapping between syntactic categories and semantic types is defined as a

function *type*:

$$
\begin{aligned}
&\mathit{type}(np) = e \\
&\mathit{type}(s) = t \\
&\mathit{type}(A/B) = (\mathit{type}(B) \rightarrow \mathit{type}(A)) \\
&\mathit{type}(B\backslash A) = (\mathit{type}(B) \rightarrow \mathit{type}(A))
\end{aligned}
$$

Syntactic backward and forward application corresponds to functional application. The final result would be the this:

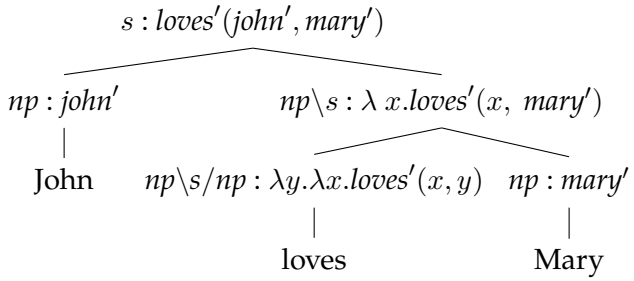

Figure 2.2: A syntactic tree for *John loves Mary*. Lexicon assigns categories to words: *John* is *np*, loves is $np\backslash s/np$ and Mary is *np*. Backward and forward composition rules derive the syntactic tree.

Given a suitable pairing between a syntactic grammar, semantic representations and corresponding general combinatory operators, this can produce structured sentential representations with broad coverage and good generalisability (see e.g. **?**). This logical approach is extremely powerful because it can capture complex aspects of meaning such as quantifiers and their interaction (see e.g. **?**), and enables inference using well studied and developed logical methods (see e.g. **?**).

2.3 Distributional hypothesis

However, such formal approaches are less able to express *similarity* in meaning. We would like to capture the intuition that while *John* and *Mary* are distinct, they are rather similar to each other (both of them are humans) and dissimilar to *dog*, *pavement* or *idea*. The same applies at the phrase and sentence level: *dogs chase cats* is similar in meaning to *hounds pursue kittens*, but less so to *cats chase dogs* (despite the lexical overlap).

Distributional methods provide a way to approach this problem. By representing words and phrases as vectors or tensors in a (usually high dimensional) vector space, we can express similarity in meaning via a suitable distance metric within that space, and can also express composition via suitable linear algebraic operations.

2.3.1 Co-occurrence based word representation

One way to produce such representations is to directly exploit Harris (1970)'s intuition that semantically similar words tend to appear in similar contexts. We can construct a vector space in which the dimensions correspond to contexts, usually other words. The word vector components can then be calculated from the frequency with which the word co-occurred with the corresponding contexts in a predefined window.

	philosophy	book	school
Mary	0	10	22
John	4	60	59
girl	0	19	93
boy	0	12	146
idea	10	47	39

Table 2.1: Word co-occurrence frequencies extracted from the BNC.

Table 2.1 shows 5 3-dimensional vectors for the words *Mary*, *John*, *girl*, *boy* and *idea*. The words *philosophy*, *book* and *school* label vector space dimensions. As the vector for *John* is closer to *Mary* than it is to *idea* in the vector space, we can say that *John*'s contexts are similar to *Mary*'s (and dissimilar to *idea*'s), therefore *John* is semantically more similar to *Mary* than to *idea*.

Many variants of this approach exist: performance on word similarity tasks has been shown to be improved by replacing raw counts with weighted values (e.g. mutual information) – see **?** and below for discussion, and Kiela and Clark (2014) for a detailed comparison.

2.3.2 Neural word embedding

Deep learning techniques use this distributional hypothesis differently. Instead of relying on observed co-occurrence frequencies, a neural model is trained to maximise some objective function related to e.g. the probability of observing the surrounding words in some context Mikolov et al. (2013a):

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t) \tag{2.1}$$

Maximising this function produces vectors which maximise the conditional probability of observing words in a context around the target word w_t, where c is the size of the training context, and $w_1 w_2, \cdots w_T$ is a sequence of training words. They therefore capture the distributional intuition and can express degrees of lexical similarity.

However, they have also proved successful at other tasks Mikolov et al. (2013b); the vectors obtained encode not only attributional similarity (similar words are close to each other), but also relational similarities **?**. For example, it is possible to extract the singular:plural

relation (*apple*:*apples*, *car*:*cars*) using vector subtraction:

$$\overrightarrow{apple} - \overrightarrow{apples} \approx \overrightarrow{car} - \overrightarrow{cars}$$

also semantic relationships are preserved:

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

allowing the formation of analogy queries similar to $\overrightarrow{king} - \overrightarrow{man} + \overrightarrow{woman} = ?$, obtaining $\overrightarrow{queen}$ as the result.[1]

Both neural and co-occurrence-based approaches have advantages over classical formal approaches in their ability to capture lexical semantics and degrees of similarity; their success at extending this to the sentence level, and to more complex semantic phenomena, depends on their applicability within compositional models.

2.4 Composition

Methods based on this distributional hypothesis have recently been applied to many tasks, but mostly at the word level: for instance, word sense disambiguation **?** and lexical substitution **?**. They exploit the notion of similarity which correlates with the angle between word vectors **?**. *Compositional* distributional semantics goes beyond the word level and models the meaning of phrases or sentences based on their parts. **?** perform composition of word vectors using vector addition and multiplication operations. The limitation of this approach is the operator associativity, which ignores the argument order, and thus word order. As a result, "*John loves Mary*" and "*Mary loves John*" get assigned the same meaning.

Concretely, if *John*, *Mary* and *loves* meaning is represented as vectors $\overrightarrow{john}$, $\overrightarrow{mary}$ and $\overrightarrow{loves}$, the meaning of the sentence *John loves Mary* is $\overrightarrow{john} + \overrightarrow{loves} + \overrightarrow{mary}$.

The To capture word order, various approaches have been proposed. **?** extend the compositional approach by using non-associative linear algebra operators as proposed in the theoretical work of **?**.

The functional application of semantic term can be replaced with tensors **?**. Then, a transitive verb is represented by matrix, which can be obtained from a corpus using the formula $\sum_i \overrightarrow{s_i} \otimes \overrightarrow{o_i}$ (the relation method of **?**), where $\overrightarrow{s_i}$ and $\overrightarrow{i_i}$ are the subject–object pairs of the verb.

The vector of the whole sentence is $\overrightarrow{loves} \odot (\overrightarrow{john} \otimes \overrightarrow{mary})$.

2.5 The tasks that require similarity

[1] Levy et al. (2014) improved Mikolov et al. (2013b)'s method of retrieving relational similarities by changing the objective function and improved the state-of-the-art results both for neural embeddings and co-occurrence based vectors.

Dialogue act tagging There are many ways to approach the task of dialogue act tagging **?**. The most successful approaches combine *intra*-utterance features, such as the (sequences of) words and intonational contours used, together with *inter*-utterance features, such as the sequence of utterance tags being used previously. To capture both of these aspects, sequence models such as Hidden Markov Models are widely used **??**. The sequence of words is an observable variable, while the sequence of dialogue act tags is a hidden variable.

However, some approaches have shown competitive results without exploiting features of inter-utterance context. **?** concentrate only on features found inside an utterance, identifying ngrams that correlate strongly with particular utterance tags, and propose a statistical model for prediction which produces close to the state of the art results.

The current state of the art **?** uses Recurrent Convolutional Neural Networks to achieve high accuracy. This model includes information about word identity, intra-utterance word sequence, and inter-utterance tag sequence, by using a vector space model of words with a compositional approach. The words vectors are not based on distributional frequencies in this case, however, but on randomly initialised vectors, with the model trained on a specific corpus. This raises several questions: what is the contribution of word sequence and/or utterance (tag) sequence; and might further gains be made by exploiting the distributional hypothesis? What is the contribution of utterance meaning to its tag?

Paraphrase detection Microsoft paraphrase corpus **?** is a collection of sentences labeled whether one is a paraphrase of another.

Disambiguation The transitive verb disambiguation dataset[2] described in **?**. The dataset consists of ambiguous transitive verbs together with their arguments; landmark verbs, which identify one of the verb senses; and human judgements which specify the similarity to the landmarks of the disambiguated sense of the verb in the context given. This is similar to the intransitive dataset described in **?**.

Consider the sentence *"system meets specification"*; here, *meets* is the ambiguous transitive verb, and *system* and *specification* are the arguments in this context. Possible landmarks for *meet* are *satisfy* and *visit*; for this sentence, the human judgements show that the disambiguated verb meaning is similar to the landmark *satisfy*, and less similar to *visit*.

The task is to estimate the similarity of the sense of a verb in a context with a given landmark. To provide our similarity measures, we compose the verb with its arguments using one of our compositional operators, do the same for the landmark and the arguments, and compute the cosine similarity of the two vectors. To evaluate performance, we average the human judgements for the same verb, argument and landmark entries, and use the average values to calculate the correlation. As a baseline, we compare to the correlation achieved using only the verb vector, without composing with its arguments.

[2]This and the sentence similarity datasets are available at http://www.cs.ox.ac.uk/activities/compdistmeaning/

Sentence similarity The transitive sentence similarity dataset described in **?**. The dataset consists of transitive sentence pairs and a human similarity judgement. The task is to estimate similarity between two sentences.

Bibliography

Ulrike Hahn. Similarity. *Wiley Interdisciplinary Reviews: Cognitive Science*, 5(3):271–280, 2014. ISSN 1939-5086. doi: 10.1002/wcs.1282. URL http://dx.doi.org/10.1002/wcs.1282.

John R. Firth. A Synopsis of Linguistic Theory, 1930-1955. *Studies in Linguistic Analysis*, pages 1–32, 1957.

Zellig S. Harris. *Papers in Structural and Transformational Linguistics*, chapter Distributional Structure, pages 775–794. Springer Netherlands, Dordrecht, 1970. ISBN 978-94-017-6059-1. doi: 10.1007/978-94-017-6059-1_36. URL http://dx.doi.org/10.1007/978-94-017-6059-1_36.

Hinrich Schütze. Automatic word sense discrimination. *Comput. Linguist.*, 24(1):97–123, March 1998. ISSN 0891-2017. URL http://dl.acm.org/citation.cfm?id=972719.972724.

G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, November 1975. ISSN 0001-0782. doi: 10.1145/361219.361220. URL http://doi.acm.org/10.1145/361219.361220.

Ido Dagan, Shaul Marcus, and Shaul Markovitch. Contextual word similarity and estimation from sparse data. In *Proceedings of the 31st Annual Meeting on Association for Computational Linguistics*, ACL '93, pages 164–171, Stroudsburg, PA, USA, 1993. Association for Computational Linguistics. doi: 10.3115/981574.981596. URL http://dx.doi.org/10.3115/981574.981596.

Douwe Kiela and Stephen Clark. A systematic study of semantic vector space model parameters. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pages 21–30, Gothenburg, Sweden, April 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W14-1503.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013a. URL http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT*, pages 746–751, 2013b. URL http://www.aclweb.org/anthology/N13-1090.pdf.

Omer Levy, Yoav Goldberg, and Israel Ramat-Gan. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning, Baltimore, Maryland, USA, June. Association for Computational Linguistics*, 2014. URL http://www.cs.bgu.ac.il/~yoavg/publications/conll2014analogies.pdf.