

THESIS FOR THE DEGREE OF LICENTIATE OF ENGINEERING

# Word Sense Embedded in Geometric Spaces

From Induction to Applications using Machine Learning

MIKAEL KÅGEBÄCK

Department of Computer Science and Engineering  
CHALMERS UNIVERSITY OF TECHNOLOGY  
UNIVERSITY OF GOTHENBURG

Gothenburg, Sweden 2016

Word Sense Embedded in Geometric Spaces  
From Induction to Applications using Machine Learning  
MIKAEL KÅGEBÄCK

ISBN 1652-876X

© MIKAEL KÅGEBÄCK, 2016

Thesis for the degree of Licentiate of Engineering  
ISSN 1652-876X  
Technical Report No. 156L  
Department of Computer Science and Engineering

Department of Computer Science and Engineering  
Chalmers University of Technology and University of Gothenburg  
SE-412 96 Gothenburg  
Sweden  
Telephone: +46 (0)31-772 1000

Cover:

2D projection of geometric representations of word tokens corresponding to different senses of the word type *paper*.

Chalmers Reproservice  
Gothenburg, Sweden 2016

*To Sandra and Alva.*



## ABSTRACT

Words are not detached individuals but part of an interconnected web of related concepts, and to capture the full complexity of this web they need to be represented in a way that encapsulates all the semantic and syntactic facets of the language. Further, to enable computational processing they need to be expressed in a consistent manner so that common properties, e.g. plurality, are encoded in a similar way for all words sharing that property. In this thesis dense real valued vector representations, i.e. *word embeddings*, are extended and studied for their applicability to *natural language processing* (NLP). Word embeddings of two distinct flavors are presented as part of this thesis, sense aware word representations where different word senses are represented as distinct objects, and grounded word representations that are learned using multi-agent *deep reinforcement learning* to explicitly express properties of the physical world while the agents learn to play Guess who?. The empirical usefulness of word embeddings is evaluated by employing them in a series of NLP related applications, i.e. *word sense induction*, *word sense disambiguation*, and *automatic document summarisation*. The results show great potential for word embeddings by outperforming previous *state-of-the-art* methods in two out of three applications, and achieving a statistically equivalent result in the third application but using a much simpler model than previous work.



## ACKNOWLEDGEMENTS

The writing of this thesis, and doing the research behind it, has been one of the most rewarding experiences so far in my professional career, a fact that is to a large extent due to the people that I had the pleasure of sharing this time with. Hence, I would like to take this opportunity to extend my warmest *thank you*.

First to my main supervisor Devdatt Dubhashi, thank you for giving me this opportunity, your guidance and inspiration, and prompt feedback 24 hours a day (yes, even in the middle of the night, this happened, not sure how you pull it off). My co-supervisors Richard Johansson and Shalom Lappin for welcome input and guidance, and Gerardo Schneider for keeping me on the narrow path in the role of examiner. Further, I would like to extend my warmest gratitude towards Richard Socher for taking the time and effort to travel here from San Fransisco and leading the discussion during my licentiate seminar.

Next my room mates, with whom I have had countless discussions on every subject worth discussing, Fredrik D. Johansson you have been a great colleague and I have learned a lot from our hours in front of the whiteboard, and Olof Mogren who co-authored my first paper and share my passion for Deep Learning. My friends and colleagues, Nina Tahmasebi for trying to teach me how to write a scientific paper, Hans Salomonsson for coding like a ninja, my star master student Emilio Jorge, and Prasanth Kolachina for digging up those hard to find references. Further, my lab mates Chien-Chung, Christos, Aristide, Vinay, Peter, Azam, Joel, and Alexander. For helping me on the administrative side I want to thank Rebecca Cyren, Jonna Amgard and Eva Axelsson, and Peter Dybjer for signing the important papers.

Finally, all my love and gratitude goes to my wonderful family for making this dream possible.





## LIST OF PUBLICATIONS

This thesis is based on the following manuscripts.

- Paper I** M. Kågebäck, F. Johansson, R. Johansson, and D. Dubhashi (2015). “Neural context embeddings for automatic discovery of word senses”. *Proceedings of NAACL-HLT*, pp. 25–32
- Paper II** E. Jorge, M. Kågebäck, and E. Gustavsson (2016). “Learning to Play Guess Who? and Inventing a Grounded Language as a Consequence”. *NIPS workshop on deep reinforcement learning*
- Paper III** M. Kågebäck and H. Salomonsson (2016). “Word Sense Disambiguation using a Bidirectional LSTM”. *5th Workshop on Cognitive Aspects of the Lexicon (CogALex)*. Association for Computational Linguistics
- Paper IV** M. Kågebäck, O. Mogren, N. Tahmasebi, and D. Dubhashi (2014). “Extractive summarization using continuous vector space models”. *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)@ EACL*, pp. 31–39
- Paper V** O. Mogren, M. Kågebäck, and D. Dubhashi (2015). “Extractive Summarization by Aggregating Multiple Similarities”. *Proceedings of Recent Advances in Natural Language Processing*, pp. 451–457

The following manuscripts have been published, but are not included in this work.

- Paper VI** N. Tahmasebi, L. Borin, G. Capannini, D. Dubhashi, P. Exner, M. Forsberg, G. Gossen, F. D. Johansson, R. Johansson, M. Kågebäck, O. Mogren, P. Nugues, and T. Risse (2015). Visions and open challenges for a knowledge-based culturomics. *International Journal on Digital Libraries* **15.2-4**, 169–187



## CONTRIBUTION SUMMARY

- Paper I** I am the main author, developed the main technical contribution, and wrote about 50% of the manuscript and experiments.
- Paper II** I initiated the project, supervised the main author, and contributed towards the manuscript (abstract, introduction, and conclusions) and the technical contribution.
- Paper III** I developed the main technical contribution and wrote 90% of the manuscript.
- Paper IV** I am the main author, developed the main technical contribution, and wrote about 80% of the manuscript and experiments.
- Paper V** I am the second author, contributed with the idea of multiplicative interaction between kernels, and wrote about 20% of the manuscript and experiments.



# CONTENTS

<b>Abstract</b>	<b>i</b>
<b>Acknowledgements</b>	<b>iii</b>
<b>List of publications</b>	<b>v</b>
<b>Contribution summary</b>	<b>vii</b>
<b>Contents</b>	<b>ix</b>
<b>I Extended Summary</b>	<b>1</b>
<b>1 Introduction</b>	<b>3</b>
1.1 Main Contributions of this Thesis . . . . .	3
<b>2 Embedding Words in Geometric Spaces</b>	<b>5</b>
2.1 Basic Vector Representations . . . . .	5
2.1.1 One-Hot Vectors . . . . .	5
2.1.2 Feature Vectors . . . . .	5
2.1.3 Bag-of-Words Vectors . . . . .	6
2.2 Dense Real Valued Vectors Representations . . . . .	6
2.2.1 CW Vectors . . . . .	6
2.2.2 Continuous Skip-gram . . . . .	7
2.2.3 Global Vectors for Word Representation . . . . .	7
2.3 Sense Aware Word Embeddings . . . . .	7
2.3.1 Instance-Context Embeddings . . . . .	8
2.4 Embedding Grounded Concepts . . . . .	8
<b>3 Applications of Word Embeddings to NLP</b>	<b>11</b>
3.1 Word Sense Induction . . . . .	11
3.2 Word Sense Disambiguation . . . . .	11
3.3 Automatic Multi-Document Summarisation . . . . .	12
3.3.1 Extractive Summarisation . . . . .	12
3.3.2 Comparing Sentences . . . . .	13
3.3.3 MULTISUM . . . . .	13
<b>4 Future Direction of Research</b>	<b>15</b>
<b>References</b>	<b>16</b>
<b>II Publications</b>	<b>19</b>



Part I  
**Extended Summary**





# Chapter 1

## Introduction

Speakers of a language tend to have a very personal relationship to the words that make up that language. Every word has a different feel to it, that somehow encapsulates the essence of what that word means, but how do you translate this feeling into a mathematical representation that can be used in computation? When engineers design communication protocols they tend to keep the symbols orthogonal and context independent which makes the protocols compact and unambiguous. However, when humans communicate they make no such effort. Instead, several words may have related or identical meaning and most words encode different senses depending on the context in which they are being used. To further add to the complexity, these idiosyncrasies follow no well defined set of rules which makes human language very difficult to comprehend in an algorithmic way.

In this thesis different ways of representing words as dense real valued vectors are studied. Starting with neural word embeddings, which are by-products from predictive modeling of word co-occurrence statistics, that are able to encode similarities between words as distances in a geometric space. Continuing with sense aware word embeddings which provide several different representations for each word, i.e. one per word sense. Finally, a first step is taken towards grounded word representations by letting agents invent their own language to communicate concepts found in images.

To provide evidence of the usefulness of neural word embeddings in real applications, a study on the effectiveness of these representations in the following applications are conducted. (1) The automatic creation of a lexicon given a text corpus which is referred to as *Word Sense Induction*(WSI), (2) the related task of *Word Sense Disambiguation*(WSD) which is the problem of assigning a sense label, from a predefined set of senses, to a word token in a text, and (3) automatic summarisation of one or more documents by picking sentences as to cover as much of the central information in the corpus as possible, i.e. *extractive multi-document summarisation*.

### 1.1 Main Contributions of this Thesis

- A method for creating sense aware word embeddings, Section 2.3.1, is presented and used to do WSI, Section 3.1, on a well known dataset achieving a 33% relative

improvement over previous *state-of-the-art* methods. For more details on these results see Paper I.

- An *end-to-end* trainable multiple-agent reinforcement learning model that invents a grounded language to play *Guess who?*. See Section 2.4 for a short introduction or Paper II for a complete description.
- A purely learned approach to WSD, Section 3.2, that achieves results on par with state-of-the-art resource heavy systems, by leveraging GloVe vectors, Section 2.2.3, and a *bidirectional long short-term memory* network. See Paper III for more on these results and a detailed description of the model.
- A study on the applicability of neural word embeddings, Section 2.2, to provide a semantically aware sentence-to-sentence similarity score for use in extractive multi-document summarisation, Section 3.3. The results are disseminated in Paper IV and Paper V.

# Chapter 2

## Embedding Words in Geometric Spaces

Representing words as vectors has many advantages, three of them being: (1) they make it possible to encode different properties of words that may be shared between words, (2) they provide well defined distance measures, e.g. euclidean distance or cosine distance, for comparing words, and (3) they can encode correlated features by using a non orthogonal basis.

### 2.1 Basic Vector Representations

Before going into word embeddings some background on traditional vector space models are given in this section.

#### 2.1.1 One-Hot Vectors

The most basic way of encoding a word in a vector space is called a one-hot encoding, i.e. a vector of the same dimensionality as the language where all but one dimension are zero and the remaining is one, the index of which encode the word type. This means that the vocabulary makes up an orthonormal basis for the vector space, which has the advantage that no assumptions about the words are being encoded in their representations. This orthonormal property makes them very useful in some applications, however, as semantic embeddings they are useless as they encode no information about the words and all words are of equal distance to each other.

#### 2.1.2 Feature Vectors

To get a more semantically meaningful representation a second approach could be to list all known features of words and let them define the basis of the space. A word representation would then be a vector of zeros and ones indicating the absence or presence

of corresponding word feature. However, this leads to a few problems. First, it is not clear that all features are equally important which means that you will have to weight these to make the geometric distance measure meaningful and weighting features by hand with no clear objective would be highly subjective. Second, it is not possible to produce the definite list of all properties of words since language is continually changing.

### 2.1.3 Bag-of-Words Vectors

In order to improve scalability and objectivity we turn to hard statistics. But how do you capture word semantics in statistics? This difficult task was answered by Harris 1954 with the *distributional hypothesis* stating that, in the words of John Rupert Firth, *You shall know a word by the company it keeps*. I.e. statistics regarding which words co-occur can be used to form word representations. An early attempt at leveraging these statistics are called *bag-of-words* representations. These representations are related to the one-hot encodings in Section 2.1.1 as they can be formed by summing the one-hot vectors of all tokens in a corpus occurring within a given context window, e.g. three words before and after, the word type to represent. If this vector is subsequently normalized to sum to one, each dimension will indicate the probability of co-occurring with the word type corresponding to that dimension. These types of representations, when trained on a sufficiently large text corpus, will be able to enjoy all three of the advantages of vector based word representations stated in the beginning of the section. However, they suffer from one crucial deficit, *the curse of dimensionality*. This is because the dimensionality of the space equals the number of words in the vocabulary, which is very high for most languages, and has been shown to render geometric distance measures ineffective for measuring similarity between words in these models (Baroni, Dinu, and Kruszewski 2014). Though a lot of effort has been spent on overcoming this limitation, via different weighting and dimensionality reduction techniques, no definite answer of how to solve the problem has been found (Baroni, Dinu, and Kruszewski 2014).

## 2.2 Dense Real Valued Vectors Representations

In an effort to overcome the dimensionality problem of bag-of-words representations, Bengio et al. 2003 introduced a new way of leveraging co-occurrence statistics by learning to predict the context surrounding the target word using a neural network. By solving this proxy problem the network is forced into assigning similar vectors, now referred to as neural embeddings, for representing similar words. This approach has many advantages, one being that the embedding dimensionality can be chosen by the user. However, this model relied on computing a distribution over all words in the vocabulary, which is too computationally expensive to train the model on a large corpus.

### 2.2.1 CW Vectors

The first practical algorithm for training neural word embeddings was instead presented by Collobert and Weston 2008. This model solved the dimensionality problem by, instead

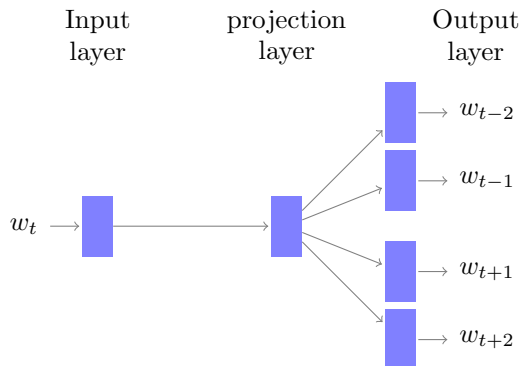


Figure 2.2.1: *The continuous Skip-gram model. Using the input word ( $w_t$ ) the model tries to predict which words that will be in its context ( $w_{t\pm c}$ ).*

of learning the probability of each word type in the context of a target word, learning to differentiate between the correct target word and a random word given a context.

## 2.2.2 Continuous Skip-gram

However, it was with the *continuous Skip-gram* model by Mikolov, Chen, et al. 2013, released within the *Word2vec* package, that neural word embeddings became widely popular. The Skip-gram model is a simplified log-linear neural network, see Figure 2.2.1, that can be efficiently trained on huge amounts of data. Later the same year this model was shown by Mikolov, Yih, and Zweig 2013 to be able to capture multiple dimensions of similarity and be used to do analogy reasoning using linear vector arithmetics, e.g.  $v_{king} - v_{man} + v_{woman} \approx v_{queen}$ .

## 2.2.3 Global Vectors for Word Representation

Though prediction based word embeddings quickly gained interest in the community and were fast replacing the counting based bag-of-words models, Pennington, Socher, and Manning 2014 showed that the two approaches had some complimentary properties and introduced *Global Vectors for Word Representation* (GloVe). GloVe is a hybrid approach to embedding words that combine a log-linear predictive model with counting based co-occurrence statistics to more efficiently capture global statistics, something they showed was lacking in the predictive models. As such, GloVe might represent the best of both worlds.

## 2.3 Sense Aware Word Embeddings

Though the word embeddings described in Section 2.2 has enjoyed much success they are actually founded on a false assumption, i.e. that each word has exactly one sense. This is clearly not true, e.g. the word *rock* may refer to either *music* or a *stone*. In this

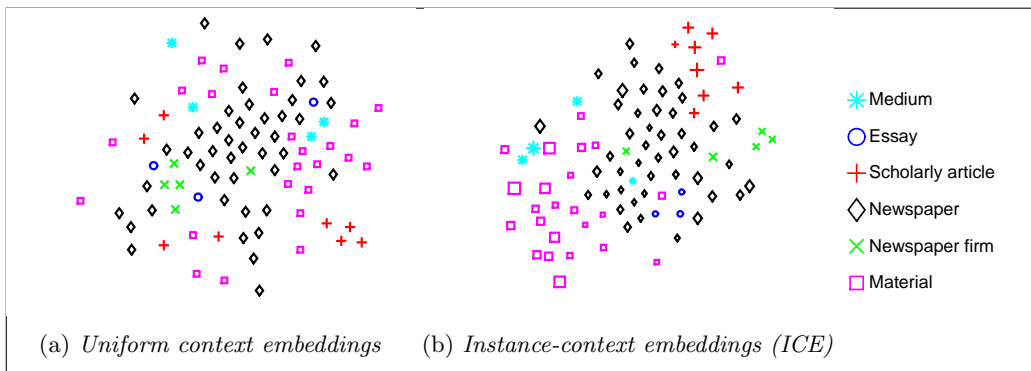


Figure 2.3.1: *Context embeddings for instances of the noun “paper” in the SemEval-2013 test data, plotted using t-SNE. The legend refers to WordNet gold standard sense labels.*

Section we turn our attention to the problem of multiple senses of a single word type. To solve this we again employ to the distributional hypothesis and use the embeddings of the surrounding words as a basis for a context specific word embedding tailored for a specific word token.

### 2.3.1 Instance-Context Embeddings

In Paper I, two approaches for computing sense aware word embeddings are introduced, where the first mainly provide a baseline for the second approach. The baseline system construct context dependent embeddings by averaging the word embeddings, described in Section 2.2.2, corresponding to the word tokens in their context. The drawback of the baseline approach, that we try to rectify in our second method *Instance-Context Embeddings* (ICE), is that it attends the same amount on all words in the context even though some words are clearly more indicative for deciding the sense of a given target word. Our solution to this problem is to attend more to the words to which the Skip-gram model assigns a high probability of occurring in the target words context. This means that the words that correlate with the target word will be attended to more, but also that very common words that correlate with every word will be weighted less. This is due to the connection between the Skip-gram objective and *pointwise mutual information* showed in (Levy and Goldberg 2014), and has the effect of creating an embedding that is more stable for words sense, see Figure 2.3.1, and less affected by the noise of unrelated words, e.g. stop words or words that are rarely used together with the target word.

## 2.4 Embedding Grounded Concepts

Another aspect that is not covered by the word embeddings described in Section 2.2 is grounding. That is, the connection between the physical world and the words in a text. Grounding is not only important as a bridge to the physical world but could also aid in the understanding of the text by using generalization of concepts via physical properties

rarely discussed in writing, e.g. that *most ground vehicles have wheels* which is apparent from images but usually not stated in written descriptions. In Paper II we take a first step towards grounded word embeddings by training agents to communicate concepts in images without any a priori shared language. i.e. they will need to create a language that encode concepts in the images in order to solve a common task. The task they are set out to solve is the game of Guess who?. A collaborative game, illustrated in Figure 2.4.1, where one player (the asking player) is tasked with figuring out which image, from a known set, that the other player (the answering player) is currently holding. To do this the asking player gets to ask questions to which the answering player will respond yes or no.

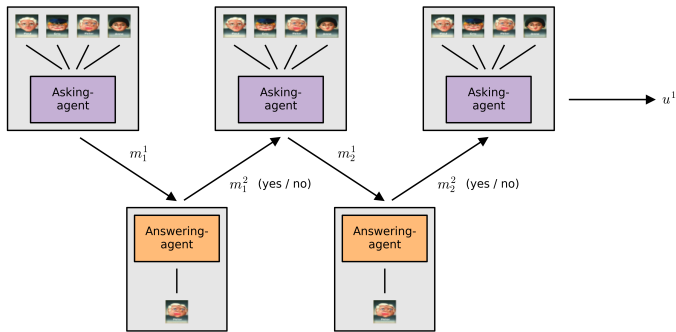




















Figure 2.4.1: *Schematic illustration of our version of the Guess Who? game.*

To get a feeling for what concepts the agents decide to encode in their words we analyzed their interactions from a restrictive setup where the agents are only allowed two words (or questions) and the set of images only consist of two images sampled from the full set. The interactions from three such setups are tabulated in Table 2.4.1 where it can be seen that question B encoded a concept, perhaps the lack of mustache, that separated the images in the first and third setup. A deeper analysis of the result is given in Paper II.

Table 2.4.1: Final message protocols between the asking-agent and the answering-agent depending on the images the agents see.

Asking-agent	Answering-agent	Message protocol		Guess	Reward
		Question	Answer		
		B	yes		1
		B	no		1
		B	yes		1
		B	no		1
		A	no		0
		A	no		1
		A	no		0
		A	no		1
		B	yes		1
		B	no		1
		B	yes		1
		B	no		1



## Chapter 3

# Applications of Word Embeddings to NLP

Though interesting in themselves, the main reason for the surge of interest in word embeddings is their applicability in *natural language processing*(NLP). Within this thesis three basic NLP application areas have been studied, and descriptions of each of them will follow.

### 3.1 Word Sense Induction

The first application considered is *Word Sense Induction*(WSI), the task of automatically creating a word sense inventory, i.e. lexicon, given a corpus. WSI is becoming an increasingly important tool for lexicographers trying to keep up with the ever increasing pace of language change. Our approach follows the work of Schütze 1998 by employing *context clustering*, i.e. embedding the context of tokens corresponding to a given word type and clustering them to find the different word senses. Traditionally the embeddings used have been different variations of the baseline system described in Section 2.3.1, i.e. bag-of-words representations. However, in paper Paper I we show that our proposed ICE embeddings, also described in Section 2.3.1, outperforms the traditional embeddings and achieved a relative improvement over the previous state-of-the-art method of 33% on the WSI task of SemEval-2013.

### 3.2 Word Sense Disambiguation

The problem of assigning a word sense, from a set of predefined senses, to a word token is referred to as *Word Sense Disambiguation* (WSD). Traditionally WSD has been approached by modeling a fixed context window surrounding the target word, i.e. the word to disambiguate, as an unordered set. Though this may work for a large set of instances it is not difficult to find examples where the order is helpful, or even necessary,

for correct disambiguation.

In Paper III a sequence modeling approach is instead taken, where the order of words play an important part, and where the window is implicitly learned during training instead of defined a priori. See Figure 3.2.1 for an illustration of the model architecture.

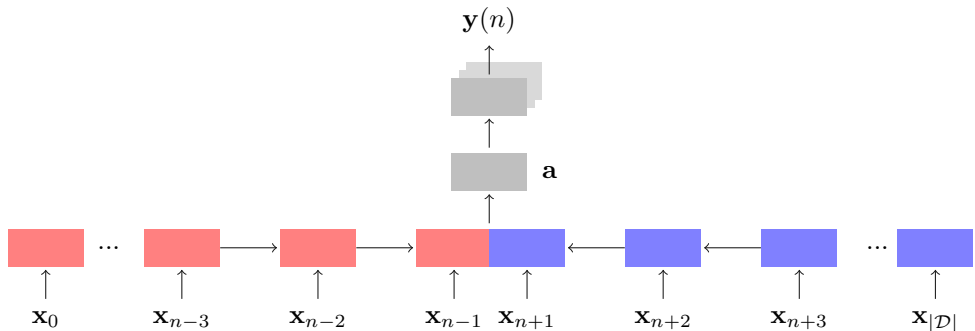


Figure 3.2.1: A BLSTM centered around a word at position  $n$ . Its output is fed to a neural network sense classifier consisting of one hidden layer with linear units and a softmax. The softmax selects the corresponding weight matrix and bias vector for the word at position  $n$ .

The model stand in stark contrast to previous work in that it relies on no external features, e.g. part-of-speech taggers, parsers, knowledge graphs, etc., but still delivers results statistically equivalent to the best state-of-the-art systems. Further, we show that word embeddings play an essential role for the performance when trained on a limited amount of sense labeled data.

### 3.3 Automatic Multi-Document Summarisation

The amount of text being produced every day has exploded, which, if you want to follow what is being written on some topic is both a blessing and a curse. A blessing in that a much richer picture is being painted, less exposed to the subjective opinions of a few writers and able to cover more aspects in-depth. This sounds great, however, humans have a limited ability to read massive amounts of text, which means that you either have to limit yourself to the opinions of a handful producers or read a fair summary. However, manually producing such a summary is in most cases prohibitively expensive which is why automatic summarisation systems are becoming an increasingly important tool to keep up with the world.

#### 3.3.1 Extractive Summarisation

Automatic summarisation comes in two distinct flavors, abstractive and extractive. Abstractive summarisation is the more general solution where an abstract representation of the documents is created and the summary is generated based on this representation. In

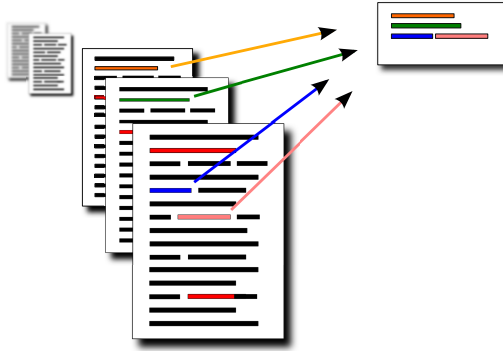


Figure 3.3.1: *Illustration of Extractive Multi-Document Summarisation.*

contrast, extractive summarisation picks the most important sentences from the documents and put them together to form the summary, See Figure 3.3.1. Though abstractive summarisation more resemble how humans summarise text, extractive summarisation has so far been more successful at solving the task.

### 3.3.2 Comparing Sentences

Using the extractive summarisation framework presented by Lin and Bilmes 2011 provides a way of extracting sentences that are both descriptive of the document set, but also diverse within the set of extracted sentences to cover as much of the information contained in the documents as possible. However, in order to perform well, this system depends on having access to a high quality sentence-to-sentence similarity measure. In Paper IV we show that word embeddings can be used to compare sentences and provide a semantically meaningful sentence-to-sentence similarity score, but to do this we have to merge word embeddings into a sentence embedding. For this we evaluate two approaches: The first is to average the embeddings of all words in the sentence and use this as a representation. The second approach use a recursive auto encoder (RAE), proposed by Socher et al. 2011 and depicted in Figure 3.3.2, to recursively merge embeddings guided by a parse tree and finally using the root layer as a sentence representation.

### 3.3.3 MULTISUM

In Paper V we follow a similar strategy but the information from the word embeddings are combined with other measures and achieve a statistically significant improvement over the state-of-the-art on the well known dataset of *Document Understanding Conference* (DUC) 2004.

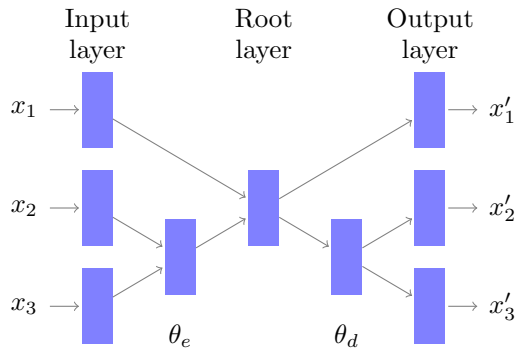


Figure 3.3.2: *The structure of an unfolding RAE, on a three word phrase  $([x_1, x_2, x_3])$ . The weight matrix  $\theta_e$  is used to encode the compressed representations, while  $\theta_d$  is used to decode the representations and reconstruct the sentence.*

## Chapter 4

# Future Direction of Research

As the licentiate thesis, to a large extent, represent a milestone on the way to a PhD, some thoughts on current and future work that will lead up to the dissertation are presented next. The general direction that is being taken is towards sequences of words and emergent properties captured through the interaction between agents. At the time of writing, this translates to the following list of ongoing projects:

**Symbolic input sequence optimization** Taking an optimization approach to the sequence to sequence decoding problem by utilizing the gradient to do optimization over a one-hot input space.

**Grounded word embeddings of human language** Connecting the grounded embeddings described in Section 2.4 with existing human language, to learn grounded embeddings of real words.

**Waveform translation** Realizing that the models behind neural machine translation are independent of the underlying data, we try to connect the spectral voiceprint of the source sentence to the voiceprint of the target sentences directly. Though challenging, this approach has the potential of producing a far superior *speech-to-speech* translation system than approaches that are constraint by having to transcode the spoken language in text, since a lot of information gets lost in that step.

# References

- Baroni, M., G. Dinu, and G. Kruszewski (2014). “Don’t count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors”. *Proceedings of Association for Computational Linguistics (ACL)*. Vol. 1.
- Bengio, Y. et al. (2003). A neural probabilistic language model. *journal of machine learning research* **3**.Feb, 1137–1155.
- Collobert, R. and J. Weston (2008). “A unified architecture for natural language processing: Deep neural networks with multitask learning”. *Proceedings of the 25th international conference on Machine learning*. ACM, pp. 160–167.
- Harris, Z. (1954). Distributional structure. *Word* **10**.23, 146–162.
- Jorge, E., M. Kågebäck, and E. Gustavsson (2016). “Learning to Play Guess Who? and Inventing a Grounded Language as a Consequence”. *NIPS workshop on deep reinforcement learning*.
- Kågebäck, M., F. Johansson, R. Johansson, and D. Dubhashi (2015). “Neural context embeddings for automatic discovery of word senses”. *Proceedings of NAACL-HLT*, pp. 25–32.
- Kågebäck, M., O. Mogren, N. Tahmasebi, and D. Dubhashi (2014). “Extractive summarization using continuous vector space models”. *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)@ EACL*, pp. 31–39.
- Kågebäck, M. and H. Salomonsson (2016). “Word Sense Disambiguation using a Bidirectional LSTM”. *5th Workshop on Cognitive Aspects of the Lexicon (CogALex)*. Association for Computational Linguistics.
- Levy, O. and Y. Goldberg (2014). “Neural Word Embedding as Implicit Matrix Factorization”. *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., pp. 2177–2185.
- Lin, H. and J. Bilmes (2011). “A Class of Submodular Functions for Document Summarization”. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. ACL, pp. 510–520.
- Mikolov, T., K. Chen, et al. (2013). Efficient Estimation of Word Representations in Vector Space. *ArXiv preprint arXiv:1301.3781*.
- Mikolov, T., W.-t. Yih, and G. Zweig (2013). “Linguistic regularities in continuous space word representations”. *Proceedings of NAACL-HLT*, pp. 746–751.

- Mogren, O., M. Kågebäck, and D. Dubhashi (2015). “Extractive Summarization by Aggregating Multiple Similarities”. *Proceedings of Recent Advances in Natural Language Processing*, pp. 451–457.
- Pennington, J., R. Socher, and C. D. Manning (2014). Glove: Global vectors for word representation. *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2014)* **12**, 1532–1543.
- Schütze, H. (1998). Automatic word sense discrimination. *Computational linguistics* **24.1**, 97–123.
- Socher, R. et al. (2011). “Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection”. *Advances in Neural Information Processing Systems* **24**.
- Tahmasebi, N., L. Borin, G. Capannini, D. Dubhashi, P. Exner, M. Forsberg, G. Gossen, F. D. Johansson, R. Johansson, M. Kågebäck, O. Mogren, P. Nugues, and T. Risse (2015). Visions and open challenges for a knowledge-based culturomics. *International Journal on Digital Libraries* **15.2-4**, 169–187.