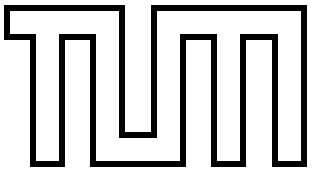


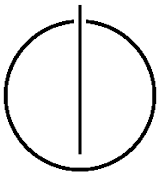
DEPARTMENT OF INFORMATICS

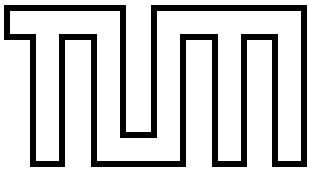
TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Information Systems

Literature Study: Use Case Analysis for Word Embeddings

Nicolas Thule





DEPARTMENT OF INFORMATICS

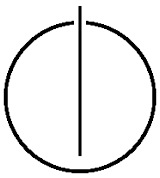
TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Information Systems

Literature Study: Use Case Analysis for Word Embeddings

Literaturstudie: Analyse von Anwendungsfällen von Word Embeddings

Author:	Nicolas Thule
Supervisor:	Prof. Dr. Florian Matthes
Advisor:	Jörg Landthaler
Date:	March 15, 2017



Ich versichere, dass ich diese Bachelorarbeit selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 14. März 2017 Nicolas Thule

Abstract

Word Embeddings are a major trend in the Natural Language Processing (NLP) community. Recent years showed a phenomenal explosion of the literature corpus in the research field. Motivated by these indicators to clarify the meaning of this trend and studying the utilization of Word Embeddings and their different Use Cases, quickly a major drawback of the vast and steadily growing amount of literature in the research field revealed itself. The body of literature lacks of comprising introductory overviews of the methods and concepts Word Embeddings are researched for. Trying to fill this gap, this thesis additionally introduces a novel layer model as a first step to an overall tool helping beginners in the field to gain fundamental insights to methods and NLP tasks leveraging Word Embeddings. This work conducted an analysis and classification approach on a literature snapshot based on a greatly researched, discussed and adapted publication which is often mentioned as the initializing starting point of the advent of Word Embeddings. Evaluation of the literature snapshot built the basis for the developed layer model and shed some light on interesting insights, gathered throughout the developments process of this thesis. Presenting possible improvements for the layer model and future directions, the thesis concludes with a short summarization of the contributed contents.

Contents

Part I.

Introduction and Methodology

1. Introduction

Word Embeddings emerged from the Natural Language Processing (NLP) research field which is an intersection of Computer Science, Artificial Intelligence, Machine Learning and computational linguistics with a long history (Chopra et al., 2013). Ever since computers support humans in the processing of text, e.g. by means of text processing tools (editor tools), people have strived to enable computers to understand natural language in a semantic way. The term NLP encompasses a large and manifold bouquet of different Use Cases in different domains, as well as a multitude of different methods and techniques. All of these different Use Cases and techniques are based on a central aspect of NLP: Natural Language has to be transformed in a way that computers can deal efficiently with text. Traditionally, a dictionary is built and text is represented in a one-hot representation. Word Embeddings are a novel and different way to perform such a transformation. Though theoretical aspects date back to the middle of the 20th century, a recently proposed method to calculate Word Embeddings gained a lot of attraction in the research community. Therefore, this thesis shall help to understand the different applications and technical approaches to utilize Word Embeddings.
The purpose of the following chapter is to outline a short introduction on Word Embeddings, provide some interesting reading material for beginners in the field, present the research goals and questions and to lay out the roadmap of this thesis.

1.1. Word Embeddings

Word Embeddings, as the understanding is of today, is a combined method of various NLP techniques emerging from different fields. From the computational linguistics field distributional semantics, which is the computationally implementable theory of meaning, established distributional semantic models (Sahlgren, 2008). Distributional semantic models assume, that the meaning of a word can be inferred by its usage, in other words: its distribution in text (Mitchell and Lapata, 2010). The field especially provided the underlying and well known *distributional hypothesis:* that words occuring within similar contexts are semantically similar (Sahlgren, 2008). Through statistical analysis of co-occurence of words and their context, these models build semantic representations in the form of high dimensional vector spaces, also known as semantic space models or (distributional) vector space models (Mitchell and Lapata, 2010). Models using co-occurence for vector representations are also referred to as count based models. Traditionally, vectors are obtained as so-called 1-of-n or on-hot representations, which basically is a matrix vector distinguishing the unique identifier of the word with a 1 from every other word in the vocabulary displayed as 0 in the vector (Goldberg, 2016). The dimension of the vector is equal to the size of the underlying vocabulary (Goldberg, 2016). Despite the wide historical implementation and developments of this approach, major problems are the computability of

the dimensionality and the sparseness of the vectors, leading to low context information for further usability (Goldberg, 2016).
Improvements in the research field of Machine Learning and especially in the computability of neural networks led to research leveraging these networks as a classifier of word contexts for an input text, known as neural Word Embeddings. Neural networks embed words and context to a low dimensional space which inherit denser context information and advantages in computability (Mikolov et al., 2013d). Neural Word Embedding models are also referred to as context-predicting models (Baroni et al., 2014). Bengio and colleagues coined the term Word Embeddings with their research, combining a neural network and a statistical language model (Bengio et al., 2003). First successes with Word Embeddings were made by Collobert and Weston (Collobert and Weston, 2008). They refined the neural network architecture and proved that Word Embeddings can improve NLP downstream tasks (Collobert and Weston, 2008).
The overall breakthrough of Word Embeddings were made by Mikolov and colleagues in 2013 (Mikolov et al., 2013a) with a simple and therefore efficient neural network structure. The light structure of the neural network made it possible to compute huge amounts of text data in a short amount of time. The researchers established two models named Continuous Bag of Words (CBOW) and Skip-gram. Both models leverage a lightweight neural network for the computation of Word Embeddings. The CBOW architecture tries to predict a word, given its context and the Skip-gram architecture vice versa tries to predict the context of a given word. These two models became highly popular in the recent years, showing several improvements of NLP tasks throughout the community. Mikolov et al. provided an implementation of their models called Word2Vec as a simple and easy to use tool [1] which received great attention and acceptance throughout the research field and led to some interesting further developments.
One year after the publication from Mikolov and colleagues and motivated by the early success of the models, Baroni at al. studied a rather superficial but interesting comparison between count based models and predictive models (Baroni et al., 2014). The result was a surprising triumph of a context predicting model based on Word2Vec (Baroni et al., 2014). Word embeddings and especially the models published by Mikolov et al. gained a lot of attention in the past few years (Li et al., 2015). The rising popularity of the models conducted the basis for this thesis as described in the following section.

Interesting literature for beginners in the field
Despite the huge amount of literature in the research community, there are rather few introductory publications focusing on Word Embeddings, their technical processes and surroundings. The publications by Mikolov and colleagues (Mikolov et al., 2013a),(Mikolov et al., 2013c) describing their methods are in some way cryptical and for beginners in the field a major drawback. Explanations on the technical aspects behind the article were published by Rong (Rong, 2014) as well as Goldberg and Levy (Goldberg and Levy, 2014). Levy and Goldberg did quite some research on the Word2Vec tool and its methods. Further publications by them (Levy and Goldberg, 2014b) and (Levy et al., 2015) are worthwhile reading material.
Several publications were provided by the research community trying to introduce and ex-

[1] https://code.google.com/archive/p/word2vec/

plain the mechanics behind the methods. One of the more thoroughly and well accepted introductions for beginners were published by Chris McCormick on his online blog (McCormik, 2016).
Further interesting insights in natural language processing, as well as embeddings and the mechanics behind artificial neural networks were recently published by Yoav Goldberg (Goldberg, 2016).

1.2. Research Goals and Research Questions

Monitoring the evolvements in the research field of Word Embeddings, some trends could be observed in the recent years. Most NLP tasks relating to text processing were researched with word embeddings. Especially the *set and go* implementation of the Word2Vec tool created by Mikolov and colleagues led to an explosion of research trying to exploit or advance the tool for improvements in NLP research. Since the publication of the Word2Vec tool and the early success, the idea of Word Embeddings and the utilization possibilities are the basis for discussions in a huge part of the related communities. The explosion also resulted in a significant increase of research, referencing the work of Mikolov et al. (Mikolov et al., 2013a). Figure 1.1[2] presents the trend of citation counts on the article from April 2016 to March 2017, observed from Google Scholar:

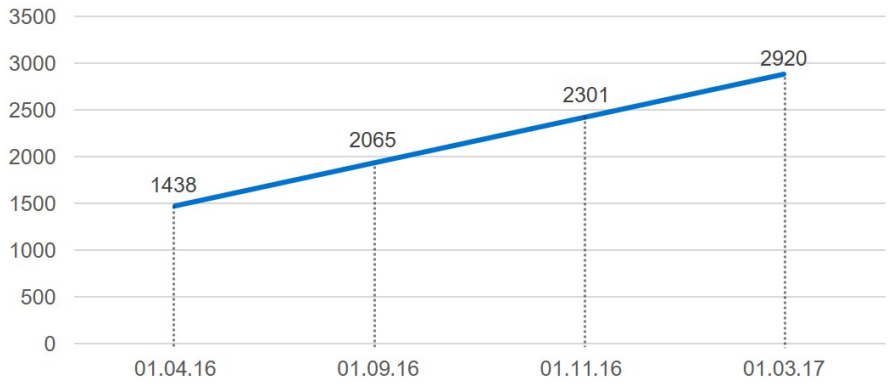


Figure 1.1.: Mikolov et al. (Mikolov et al., 2013a) citation count trend. The figure displays several observed citation counts from Google Scholar and illustrates the overall uptrend. X-axis dates of observations. Y-axis number of citations

The displayed figure includes dates on which citation counts were observed on Google Scholar. The increasing rate of citations on such a recently published paper in this short amount of time supports the assumption that the underlying methods led to a breakthrough in the field. The trend also suggests that the research conducted by Mikolov et al. is of increasing importance for the community and a huge amount of researchers are interested in the topic. Additionally it serves as a representation of the initial starting point of this thesis, focusing the analysis on the trend and trying to clarify its meaning.

[2] confirming screenshots in Appendix B

Acknowledging the important role of natural language processing in the field of computer science and the emerging trends in research related to Word Embeddings, the focus and interest of this thesis was limited to research on Word Embeddings, regarding organizational, (social) media, educational and legal environments. This limitations combine research interests based on the study field of Information Systems the thesis is located in and influences from the research focus of the conducting chair.
The following research questions[3] are the basis for this thesis:

Q1. How can Use Cases for Word Embeddings be categorized?

Q2. Which Use Cases that apply to organizational context can be identified in the existing body of literature?

Q3. For the most interesting Use Cases: How do they work technically?

Q4. What are the most frequent Use Cases?

Q5. What further technological developments of Word2Vec can be identified in the body of literature?

Despite the vast amount of research conducted in the field of Word Embeddings, the major challenge for beginners in the research field is to gain a structured conceptual overview of methods and tasks Word Embeddings are researched for. Therefore, a goal of this thesis is to introduce readers working with or interested in Word Embeddings to a layer model. It serves as an overview on different NLP tasks and domains Word Embeddings are researched and applied for.
During the process of this thesis, the focus shifted from mainly a literature analysis to the development of the layer model and it turned out as one of the main contributions this thesis has to offer. The structure of the model is described in Chapter 3 and introductions to the different layers in Chapter 4.

[3] A definition of the contextual understanding for the term Use Case will be given in Section 2.1

1.3. Structure of the Thesis

The first chapter outlines a short introduction on Word Embeddings, points out some interesting reading material for beginners in the field and presents the research goals of this thesis. Chapter 2 describes the methodology for finding specific literature, setting the scope for the thesis and presents the classification approach. A conceptual description of the developed layer model is presented in chapter 3. Chapter 4 outlines the content of the layer model based on the classification approach. It concludes with an evaluation of the model by describing a selected Use Case to illustrate the models purpose. Chapter 5 displays the results of the statistical evaluation on the literature snapshot. Following the emerging trend in the field of Word Embeddings, Chapter 6 gives a short overview on selected developments based on Word2Vec, found in the literature corpus. Outlining Lessons Learned, Chapter 7 presents insights and findings relevant for this thesis. Chapter 9 presents future research and possible directions. Finally Chapter 9 summarizes the main concepts and research results of the thesis.

2. Methodology

The research philosophy and strategy highly depends on the underlying goals. Following the research questions and objectives for this thesis outlined in Chapter 1, the purpose of this chapter is to describe the research limitations, the scope of the thesis and the approach on how the literature was classified.

2.1. Research Basis

Word Embeddings
The term Word Embeddings defined by Mikolov et al. as continuous vector representations of words (Mikolov et al., 2013a), can be understood as continuous word vectors and all terms are equally used in this thesis.
The literature corpus comprises several further terms for Word Embeddings mostly equally used. Among others continuous-valued word embeddings (Mnih and Kavukcuoglu, 2013), neural embeddings (Levy et al., 2014) and distributed word representations (Bansal et al., 2014).

Definition Use Case
The employed term Use Case, well known in the computer science field as a description between a system and its environment (Fantechi et al., 2003), is slightly different defined for this thesis. Based on the research interest of the utility of Word Embeddings, the term Use Case is understood as research on Word Embeddings or actual applications in organizational contexts. This does not limit the examination of the literature solely to organizational environments, due to the fact that an additional tendency of the research interest was to understand the utilization of word embeddings in general.

Limitations
As elucidated in the introductory chapter, the observed citation count uptrend on the article published by Mikolov and colleagues (Mikolov et al., 2013a) built the starting point for this thesis.
The limitation to Google Scholar resulted from comparing the citation counts with Scopus[1], CiteSeerx[2] and Semanticscholar[3], resulting in the observation that Google Scholar had more than twice the number of citations listed than every of the other examined databases. The quality of the citations varies, depending on the citation database. Especially Google Scholar tends to list a lot more low quality publications like theses, reports and other non journal publications. This flaw is not always negative for researchers, since

[1] https://www.scopus.com/
[2] http://citeseerx.ist.psu.edu/index
[3] https://www.semanticscholar.org/

it presents possibilities to examine a broader range of research, also conducted by the research community of the specific field. Former studies like (Meho and Yang, 2007) show that Google Scholar tends to have explicit advantages due to the stated facts. These conditions were the reason, the initial literature elicitation was solely based on Google Scholar.

2.2. Scope of the Thesis

The snapshot of the citation literature was conducted on October 01, 2016. The extraction of the citation list from Google Scholar was executed with the Publish or Perish tool[4]. After a short lookup process of the specified article, the software is able to query Google Scholar for the citation list. The used tool retrieves by default just the first 1000 publications citing the article. This limitation was used for a pre-examination for the evaluation if a retrieval of further publications is necessary. The retrieved list can be easily copied with various formats for further processing in different tools. For the classification process the list was exported to Excel. After cleaning the list from duplicates resulting in **989** publications, the list was shortly assessed for the quality of the citations. The result of the assessment showed decreasing quality and usefulness of the publications from the beginning to the end of the list, including non-English articles and articles rarely or not cited in combination with older dates of publication. This indications led to the assumption that further citations would not add value to the examination of the literature corpus and therefore set the scope of the classification approach.
Solely concentrating on the evaluation of the literature snapshot, it quickly displayed its limitations for the development of the thesis. Evolving from an analysis of the list to an additional conceptual model serving as a literature and later method overview, further research had to be done. The purpose of the additional research was to extend the model approach and to verify its meaningfulness. The additional literature search was not limited to Google Scholar and aimed to find appropriate research based on the respective topic.

2.3. Classification Approach of the Literature

The classification approach tries to answer research question **Q1.** *How can Use Cases for Word Embeddings be categorized?* and was primarily focused on classifying every publication from the literature snapshot into its respective domain. The underlying goal based on research question **Q2.** *Which Use Cases that apply to organizational context can be identified in the existing body of literature?*, was to separate research relating to organizational context for further evaluation. The classification of the 989 publications was a iterative process with several improvements since the domain clusters evolved from the process and several publications are classifiable to more than one domain. The resulting domain clusters partitioned into Enterprise, Media/Social Media, Education, Legal, Stocks, Basic Research, Bio/Medicine, Image, Video, Speech, Language and AI. For a short description and their including research please refer to section 3.4. A statistical analysis and the distribution of the domains is displayed in chapter 4.

[4]http://www.harzing.com/resources/publish-or-perish

Deeper inspections of the publications based on research questions **Q2.** and **Q3.**, led to the differentiation of several additional layers of the later developed model. Based on the huge amount of different NLP tasks Word Embeddings are researched for and to gain a deeper understanding of the fundamental methods, further literature research was conducted.
The resulting differentiation of approaches and tasks built the basis for the presented layer model introduced in chapter 3.
Word vectors are mostly used as input for several downstream tasks in a NLP pipeline (Nadkarni et al., 2011). A downstream task utilizes the methods of an underlying task (Nadkarni et al., 2011). A different view often used in the literature is that word vectors are a mere pre-processing step for further tasks (Nadkarni et al., 2011). A NLP pipeline is a combination of several tasks and downstream tasks serving the purpose of a NLP application (Nadkarni et al., 2011). Therefore the layer model introduced in Chapter 3 is developed layer-wise and can be seen as a pipeline. Extraction of common patterns in the literature snapshot and confirmation with additional literature led to the differentiation of the technical approaches, basic tasks and domain independent tasks completing the pipeline from word vectors to real word applications in specific domains.

The following chapter presents the developed layer model and short descriptions of the respective layers based on the stated approach.

Part II.

Layer Model and Classification of the Literature

3. Layer Model

Introducing a novel model, this chapter displays the structure in a layer-wise manner. The layer approach is based on the idea of a NLP pipeline, reaching from input up to the respective domain a Use Case is applied in. Starting with word vectors as input, the model provides outlines of the different technical approaches, basic and domain independent (DI) tasks up to the domains they are researched in. The structure is motivated by several aspects. A overview of technical approaches illustrates the different entry points for various pipelines. A distribution across domains illustrates the utilization and adaptability of the idea of Word Embeddings. Technical approaches and domains builds the upper and lower end of the respective pipeline and enclose basic and DI tasks in the layer model.
The structure can be displayed as followed:

Word Vectors	Layer 0
Technical Approach	Layer 1
Basic Tasks	Layer 2
DI Tasks	Layer 3
Domains	Layer 4

Figure 3.1.: Structural overview of the developed layer model

The purpose of the model is to serve as an overview of methods and tasks Word Embeddings are applied and researched for, as well as a general starting point for an analysis on Word Embedding literature and a guiding structure for reading and understanding of the topics. The huge amount of literature concerning Word Embeddings limited the selection of content for the layers of the model. Future research will be conducted to further develop the layers and broaden the range of content. A selected Use Case example explained on the basis of the model is described in Section 3.5.

For a complete overview of the layer model including the analyzed layers presented in the Sections 4.1 through 4.4 refer to Appendix A.

4. Classification

This chapter describes the results of the literature classification carried out as described in the chapter methodology based on the developed layer model. It starts with a classification and description of the technical approaches of Word Embeddings. Followed by a description of basic and domain independent (DI) tasks and a description of the most common domains Word Embeddings are used in. The chapter concludes with a selected Use Case elucidating the utilization of the model.
The sections 4.1 through 4.4 refer to research question **Q1.** *How can Uses Cases for Word Embeddings be categorized?*.

4.1. Technical Approach

This section provides a short overview about the technical concepts of Word Embeddings and how they were classified in this thesis.
Research concerning Word Embeddings utilizes several technical approaches. So far there exists no illustrative comparison in the literature corpus. As a starting point for the overall development of the layer model the thesis limits the approaches to the in Figure 4.1 displayed and in this section outlined methods. Further evaluation of the literature and extension of the introductory descriptions are part of the development process in the future.
Word vectors are the input for several NLP downstream tasks and therefore illustrated as Layer 0 in the model. Figure 4.1 gives and overview of technical approaches researched with Word Embeddings, including word vectors for illustrative purposes. Operations on vectors like the computation of the cosine similarity build the basis for Layer 1. A technical approach is basically an operation on or with Word Embeddings and can be divided in Similarity, Clustering and Classification.

Similarity
Similarity regarding Word Embeddings is one of the basic concepts, methods like Word2Vec are researched for. Starting with vectors, the application of simple vector arithmetic and especially cosine similarity on word vectors is fairly easy. The cosine similarity measures the angle between two vectors and can be seen as the distance between them. The improvements of Word Embeddings as continuous vectors combined with cosine similarity led to fast and mostly accurate computation of several downstream tasks in NLP. Examples are word similarity (Mikolov et al., 2013a) or later document similarity (Mikolov et al., 2013c). Word vectors created with Word2Vec are learned unsupervised, meaning there is no human annotation of the training data involved. This also applies to other word embedding methods.

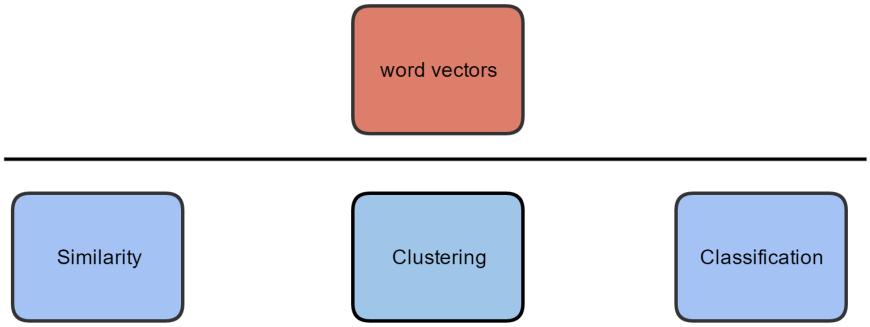


Figure 4.1.: Overview technical approaches of Word Embeddings as classified for the layer model. The figure displays Layer 0 indicated as red box and content of Layer 1 indicated as blue boxes.

Clustering
The category Clustering refers to research concerning cluster analysis on text. The objective of cluster analysis on text is to classify certain parts of text ranging from single words to whole documents into categories on the basis of their similarity (Rodriguez and Laio, 2014). Text Clustering is executed unsupervised in the majority of the literature. One of the most researched and applied method for clustering is k-means (Jain, 2010). The term document clustering is often used equivalently to text clustering in the literature.

Classification
Classification aims to classify text to one or more categories or classes (Zhang et al., 2015). Text classification is implemented as a retrieval process for labels for the given text (Zhang et al., 2015). A byproduct of this process is often a confidence measure for the label assigned to the text, to confirm the classification or for optimization of the method at training stage. Text classification helps with several downstream tasks like sentiment analysis and topic modeling (Yang et al., 2016). Document and text classification are often used equivalently in the existing literature.

4.2. Basic Tasks

Research for basic tasks as classified for this thesis focuses more on specific NLP tasks than on a specific domain. Therefore most of the research in this category could be applied to several domains depending on the downstream task or application. This category like the technical approach category consists of basic research, trying to improve efficiency of a task with Word Embeddings. This group gives a further insight on the basic concepts Word Embeddings are researched for and an introductory overview of some concepts Word Embeddings are applied to. Due to the huge amount of literature, obviously not all concepts can be pictured. The selected tasks are a mere starting point for the evolvement of the layer model. The following illustration shows the basic tasks which are explained in this section:

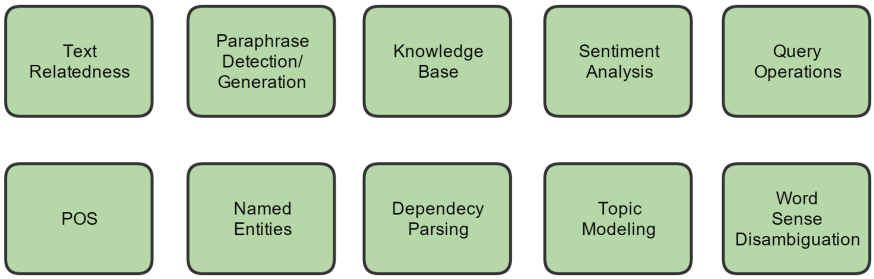


Figure 4.2.: Overview basic tasks of Word Embeddings as classified for the layer model. The figure displays Layer 2 indicated as green boxes

Text Relatedness
In the category Text Relatedness research for text similarity and analogy tasks were classified. These include: word, sentence, block, document, short text similarity or analogy.
The difference between similarity and analogy is not that obvious at first and some researchers tend to mix up the terms. Similarity is the comparison between two objects or in this context words. Words which occur in similar context are closer to each other in the embedded space (Levy et al., 2014). Analogy on the other hand is the comparison between two word pairs. One of the more famous analogy tasks in the field was researched by Mikolov and colleagues (Mikolov et al., 2013c). A better understandable explanation is provided by Levy et al.: ”In a word-analogy task we are given two pairs of words that share a relation (e.g. ”man:woman”, ”king:queen”). The identity of the fourth word (”queen”) is hidden, and we need to infer it based on the other three (e.g. answering the question: ”man is to woman as king is to - ?”).” (Levy et al., 2014).
Word analogy exploits word similarity to some extend. The relation ”man - woman + king” is executed via vector arithmetic and the resulting vector is compared to the vectors surrounded in the embedded space. The comparison results in a list of vectors which are nearest to the input relation. Mikolov and colleagues showed that the combination of vector arithmetic and a similarity measure results in the vector ”queen” as the nearest vector to the vector of the input relation (Mikolov et al., 2013c).

One major advantage with Word Embeddings is that vector arithmetic is applicable to more than the word analogy task. The composition of several word vectors can be used to compute phrase or sentence vectors as done by (Mikolov et al., 2013c). This resulted in further research concerning phrase embeddings (Mikolov et al., 2013c), paragraph and document representations (Le and Mikolov, 2014),(Gupta et al., 2016a). Driving text embedding further, Kiros et al. (Kiros et al., 2015) developed a method to embed complete sentences instead of concatenating vectors word-wise.

Paraphrase Detection/Generation
Paraphrase detection compares two text snippets and evaluates if they have the same meaning (Madnani and Dorr, 2010). This reaches from word similarity up to document similarity. The research in this category relates to paraphrase detection based on sentences which is also referred to as sentential paraphrase detection and the common understanding of paraphrase detection in the main literature pool (Madnani and Dorr, 2010). Regarding Word Embeddings, Kiros et al. apply their skip-though vectors to the paraphrase detection task and achieved promising results (Kiros et al., 2015). Paraphrase detection is similar to answer selection and could be applied alongside with it in a question answering system (Yu et al., 2014).
Paraphrase generation on the other hand tries to create a similar snippet to a given input (Iyyer et al., 2014a). This task can be useful to certain applications regarding question answering, text creation or query expansion (Zhao et al., 2009).

Knowledge Base
The category Knowledge Base pools research relating to Knowledge Bases (KBs). Examples for such KBs are DBpedia (Lehmann et al., 2015) or Freebase (Bollacker et al., 2008). These Knowledge bases contain vast amount of data. Utilization of KBs for information retrieval is an example for research conducted with Word Embeddings (Bordes et al., 2011) as well as research regarding KB expansion and generation (Dong et al., 2014).
This category is deeply linked with other NLP research fields like information extraction in general, the Open Information Extraction paradigm (Banko et al., 2007), relationship extraction (Fu et al., 2014) and ontology generation (Gupta et al., 2016b). These fields are often the basis for research on knowledge bases.

Sentiment Analysis
Sentiment Analysis aims to extract the polarity (positive vs. negative vs. neutral) of a specific object (Wang et al., 2014). The polarity of an object is highly subjective and mostly expressed in opinion-based information like reviews or social media messages (Feldman, 2013). Extraction of these opinion-based information can be valuable to customers searching for opinions on a specific product or like-wise for companies staying up-to-date on their own products or reputation throughout an internet platform (Feldman, 2013). Referring to a single subject treated in the object, sentiment analysis is executed on document (the whole text acts as the object) or sentence level (Feldman, 2013). Traditionally, text is classified in positive or negative polarity but also additional classes like neutral or numeric scales (e.g. 5 star ranking on consumer platforms) can be applied (Feldman, 2013). Aspect based sentiment analysis refers to a more detailed sentiment analysis of the object (Guha et al., 2015). One example is if a text contains more than one polarity statements e.g.

negative and positive ones for different subjects, an overall sentiment analysis could not differentiate between the subjects and eventually would conclude to a more fuzzy result (Guha et al., 2015). Therefore the sentiment analysis method has to differentiate between the subjects and extract the polarity subject-wise (Guha et al., 2015) as well as to give an overall polarity score for the text (Feldman, 2013). Before the sentiment analysis can be executed, a subject or category classification has to be applied. This subtask is highly related to text classification and therefore mostly executed supervised (Guha et al., 2015). Unsupervised sentiment analysis is rather uncommon since training data for the majority of tasks and domains is available and as well a semantic orientation of the Word Embeddings have to be calculated given predefined polarity words (Feldman, 2013).

Query Operations
The category Query Operations comprises research regarding query clustering, classification, expansion, relaxation, auto-completion and recommendation. Research on queries and especially search queries is mostly conducted for information retrieval systems or purposes. The majority of query operations are researched for improving recall and precision for better retrieval results (Manning et al., 2008). The same way text clustering and classification also *query clustering* (Kolluru and Mukherjee, 2016) and *query classification* (Yang et al., 2015) aims to find categories and classify queries into categories. These tasks are often pre-processing tasks to improve the quality of the query results. *Query expansion* is the task of enriching the seed query with relevant terms for better retrieval of documents (Zamani and Croft, 2016). Regarding Word Embeddings this task is often executed with a classification algorithm like k-nearest neighbour (Roy et al., 2016). In contrast, query relaxation aims to get better results by eliminating irrelevant results (Shi et al., 2016). *Query autocompletion* is the task of suggesting query candidates to users just after a few keystrokes (Cai and de Rijke, 2016). This often results in a candidate list proposed to the user in which the query candidates are ranked by typed input matches or by semantically related queries based on search logs (Cai and de Rijke, 2016).

POS
Part-of-speech (POS) tagging is the task of automatically assigning parts of speech like nouns and verbs to words in the given text (Fonseca et al., 2015). POS tagging mainly is a pre-processing task for many other NLP downstream tasks like Named Entity Recognition or Dependency Parsing (Maynard et al., 2016).

Named Entities
Named Entity Recognition and Classification (NERC) is concerned with identifying names of entities in text and the classification of the entities (Maynard et al., 2016). Entity recognition (NER) identifies predefined entities in text including persons, organizations or numeric expressions like time and dates (Maynard et al., 2016). Entity classification (NEC) is concerned with classifying the found entities into predefined categories (Maynard et al., 2016). Named Entity Linking (NEL) in contrast to NERC links the entities to the respective entity in a given knowledge base (Maynard et al., 2016). The output highly depends on the underlying knowledge base which often is to general or in some cases specifically constructed for the researched purpose (Maynard et al., 2016).

Dependency Parsing
Dependency parsing analyses the syntactical composition of text based on dependency representations. The most popular dependency representation of text is a parse tree also called dependency tree, which essentially is a grammatical analysis of a given sentence or text (Nivre, 2005).

Topic Modeling
Topic modeling is the task of identifying one or more latent topics in a text corpus (Arora et al., 2012). Topic models can help to identify common topics over several documents or generally can help to identify central aspects of the data (Arora et al., 2012). Topic modeling is traditionally researched with probabilistic generative models like Latent Dirichlet Allocation (LDA) and matrix factorization techniques like Non-negative Matrix Factorization (NMF) (OCallaghan et al., 2015). Recent approaches combine LDA with Word Embeddings to improve topic modeling (Moody, 2016), (Das et al., 2015), (Niu et al., 2015). Xun and colleagues (Xun et al., 2016) as well as Li et al. (Li et al., 2016a) applied Word Embeddings to topic modeling for short texts with promising results for future directions. Topic modeling is typically used in text summarization and document classification applications (Li et al., 2016b). Most of the researched models are unsupervised and so far there are no standalone breakthroughs with Word Embeddings on topic modeling in the current literature corpus.

Word Sense Disambiguation
Word Sense Disambiguation (WSD) is the task of identifying the meaning of a word used in the context of a sentence if the word has multiple meanings (polysemic) (Edmonds and Agirre, 2008). An example described in (Edmonds and Agirre, 2008) would be the meaning of *pen* depending on its used context. Pen could be the writing instrument or a "enclosure for confining livestock" and more (Edmonds and Agirre, 2008). Intuitively recognizable is that WSD is a hard problem in Natural Language Processing. In the context of Word Embeddings there are supervised (Trask et al., 2015) as well as unsupervised (Bartunov et al., 2015) methods showing promising results for future directions in WSD research.

4.3. DI Tasks

Domain independent (DI) tasks are the located on Layer 3 regarding the layer model. They leverage one or more base tasks and normally the applicability in almost every domain is very high. In some cases an overlap exists. An example could be the combination of several DI tasks to build a recommender system. Also other relations between DI tasks exist but are not further evaluated in this thesis.
The following overview shows a selection of DI tasks.

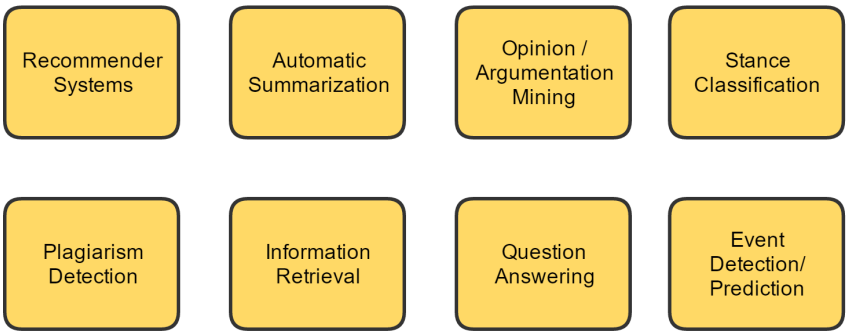


Figure 4.3.: Overview domain independent tasks of Word Embeddings as classified for the layer model. The figure displays Layer 3 indicated as yellow boxes

Recommender Systems
Recommender Systems are applications composed of several software components. Leveraging different technical approaches and base tasks the main purpose of nearly every recommender system is to ease the decision making of the systems user, enhance user experience of the implemented platform or exploit recommendations for revenue enhancements. Beyond well known and researched product recommender systems like (Barkan and Koenigstein, 2016),(Grbovic et al., 2015b) and (Wang et al., 2015), Word Embeddings are used to predict and recommend user actions on social media platforms (Ozsoy, 2016) and even for document recommendation systems used for conversational contexts (Habibi and Popescu-Belis, 2015). Research classified in this category has the aim of building recommendation applications or investigating in different tasks for improving such.

Automatic Summarization
Document summarization is an important task in the age of information. Huge amounts of information and especially in the form of text has to be processed by either humans or machines. It can be helpful for decision making, news generation or intelligence purposes to just name a few examples (Nenkova et al., 2011). The summarization task is divided in extractive and abstractive summarization (Kågebäck et al., 2014). Extractive summarization relies solely on the given document and builds the summarization by extracting keywords or sentences (Kågebäck et al., 2014). Abstractive summarization on the other hand can contain elements of the used document but overall expresses the summary in

the own words of the summary author or method (Kågebäck et al., 2014) often leveraging tasks like text paraphrasing (Rush et al., 2015).

Opinion/Argumentation Mining
Opinion mining tries to extract opinions from text. The applications reach from analyzing product reviews for recommendations or user preferences to analyzing social media posts for social movement monitoring (Ravi and Ravi, 2015). Often used with sentiment analysis, NERC and topic modeling opinion mining extracts the opinion holder, the opinion expression and the sentiment of the expression for a better classification of the analyzed text (Liu et al., 2015) and (Ma et al., 2016).

Argumentation mining in some cases is complementary to opinion mining as it extracts arguments which explain why a certain opinion is positive or negative (Moens, 2013). Not restricted to opinions, argumentation mining can also be helpful providing arguments on examined topics in discussion related environments like political discourses (Naderi and Hirst, 2014) or decision related environments (Moens, 2013).

Stance Classification
Building on opinion and argumentation mining stance classification determines whether the author of a text is in favor, against or neutral to a specific target (e.g. a users position to a product or topic in a debate) (Mohammad et al., 2016) and (Sobhani et al., 2015). Stance classification can be helpful with analyzing a users social media profile for e.g. political ideology detection (Iyyer et al., 2014c) or in debate related environments e.g. online forum discussions (Boltuzic and Šnajder, 2014).

Plagiarism Detection
Plagiarism detection is the task of automatically detecting reuse of text in a given document (Zhang et al., 2014). Mainly known in the educational domain plagiarism detection becomming more and more important due to a high reuse rate of online published text (Zhang et al., 2014). Textual similarity is one of the often used basis in plagiarism detection and gained more and more upstream through Word Embeddings in the past few years (Ferrero et al., 2017).

Information Retrieval
Information retrieval (IR) is the task of answering an information need with relevant information (Büttcher et al., 2016). The most known application is a search query retrieving information based on the query input (Büttcher et al., 2016). Modern search engines are the prime example for such a system. Beyond research relating to search queries, also research regarding applications or tasks answering informational need to the most extent, were classified in this category.

Question Answering
Question answering (QA) determines an answer to a given question or in some cases research focuses on answer selection. The latter task is especially helpful in online community platforms retrieving or labeling the best answer to a given question (Tran et al., 2015). QA in general serves well in applications reaching from information retrieval (Yu et al.,

2014) to Artificial Intelligence (Weston et al., 2015). Typical Use Cases are Chatbots (Yan et al., 2016).

Event Detection/Prediction
Event detection identifies specified types of events in text (Nguyen and Grishman, 2015). Applications for this task can be e.g. natural disaster damage assessment via social media message mining (Cresci et al., 2015) or monitoring user generated content for event detection and tracking (Atefeh and Khreich, 2015).
The term event prediction is equivalently used for establishing causal contexts of events from text (Zhao et al., 2017) and in a few but nontheless interesting cases used for stock movement prediction by leveraging event detection and causal contexts (Ding et al., 2015).

Further very interesting tasks are automatic translation (e.g. (Mikolov et al., 2013b)) located in the ”Speech” domain and multimedia information retrieval located in the ”Image” and ”Video” domains. Due to the scope of the thesis these tasks were not further investigated.

4.4. Domains

Motivated by the interest in the distribution of research regarding Word Embeddings across domains the classification of the literature regarding domains was one of the first steps towards this model. An overview of domains simplifies the evaluation of research fields interested in the idea of Word Embeddings. The most common domains are given in the following overview.

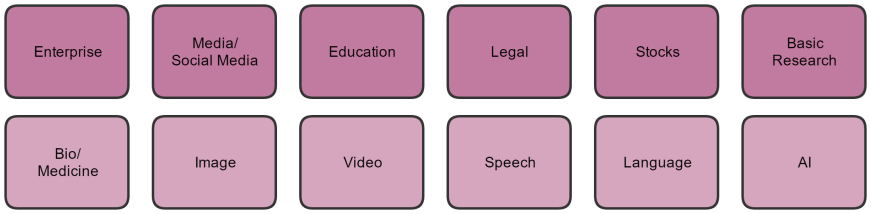


Figure 4.4.: Overview domains Word Embeddings as classified for the layer model. The figure displays Layer 4 indicated as pink boxes

The difference in the shading is due to the research focus set for this thesis. The domains colored darker are researched in more depth than the lighter ones. The huge amount of literature on Word Embeddings limited the selection of the displayed domains. Future research conducted outside of the literature snapshot and regarding little investigated domains will evolve this layer. Additionally, some research could be placed in more than one domain but were classified to the most applicable or important domain described by the respective researchers.

Enterprise
The ”Enterprise” domain reflects research which applies Word Embeddings to organizational context with a business-oriented tendency. This category includes examples which are either actually applied Use Cases or research in the organizational context which could easily be applied. Especially query operations are placed in this category since they have their origins in the online environment and are considered in this thesis as developments for enterprises. Examples for Use Cases are recommender systems, query mining and product categorization.

Media/Social Media
The ”Media/Social Media” domain includes research which applies Word Embeddings to the social media environment and general media like news papers or other print media. The distinction between textual media and non-textual media like images and multimedia is crucial in the context of Word Embeddings and Natural Language Processing and plays an important role for the readers understanding. Social media mining is the most prominent Use Case in this domain.

Education
In the ”Education” domain research on Word Embeddings is conducted for teaching, educational research and other educational purposes. Example Use Cases are automatic marking and plagiarism detection.

Legal
The ”Legal” domain encloses research relating to political as well as to juristic topics. Example Use Cases are political ideology detection and argumentation mining related to the juristic environment.

Stocks
This domain comprises research relating to stocks or stock market topics. An interesting Use Case example is stock movement prediction.

Basic Research
The domain ”Basic Research” can be understood as research not applied to a specific topic or domain as well as in more cases research regarding basic and domain independent tasks. Basic and domain independent tasks are explained in the respective sections in which also some examples are given for better understanding of the concept.

Bio/Medicine
Research conducted in the domain ”Bio/Medicine” focuses mainly on biomedical literature, clinical data and drug reaction classification in social media. The latter is one example which could be categorized in more than one domain, but in this context social media is only seen as the platform and the purpose of the research arises out of the biomedical field. The explicit label of this domain is due to no or little recognition of research concerning other natural science fields so far. Further success of Word Embeddings could lead to adaptions in the future.

Image
In the domain ”Image” research is classified which relates to scene description, visual question answering, image annotation, information retrieval related tasks like image search and other image related tasks.

Video
The domain ”video” includes research which relates mostly to action recognition of video material and video annotation.

Speech
In the domain ”Speech” publications were classified which conduct research relating to spoken language understanding. Automatic speech recognition (ASR) and conversational interaction like dialog systems are the main focus in this field. Word embeddings in this field are mainly researched to improve ASR error correction.

Language
The ”Language” category contains research focusing mainly on automatic translation and language knowledge base research. Language knowledge base research is trying to improve word sense disambiguation, paraphrase detection, word analogies, dependency parser and ontologies. Furthermore research regarding different languages than english were classified in this domain. One example is research on chinese word segmentation.

AI

The last domain "AI" is a classifier for research which concerns machine learning and comprehension in a more broader aspect regarding natural language processing as well as intelligent assistants and machine-user interaction. Research in this category is hard to define and in some cases transitions to other fields are blurred. This is due to the fact that, as stated before, some cases could be classified in more than one domain, but more important also often more than one method and domain is scratched if it comes to combining research for artificial intelligence. This domain was solely created for research which purpose or main idea is to create such AI to some extent.

4.5. Use Cases

This section elucidates the presented model with a selected Use Case. The example contours the evaluation process of literature concerning Word Embeddings and outlines the usage and usefulness of the model. The evaluation of this exemplary Use Case correlates with research question **Q3.** *For the most interesting Use Cases: How do they work technically?* and illustrate the overall classification approach of this thesis. Due to time limitations for this thesis research question **Q3.** is just partly answered, but the illustrative example illuminates the potential future direction and possibilities of the presented layer model.

Use Cases in organizational context
Regarding the classification of domains this thesis tried to set a hard line depending on the research purpose the respective publication pursued. Research question Q2. refers to organizational context which is not clearly defined by itself. The definition of the domain Enterprise were based on this context to establish a line between the research focusing on business-oriented context and others. Despite the fact that organizational context widely refers to business-oriented environments, the examination of literature for this section included the domains Media/Social Media, Education and Legal which was motivated by the research interest of this thesis. Future work providing examples located in this and other domains is planned as one of the next steps.

Use Case
This paragraph provides a description of the Use Case followed by a short evaluation leveraging the presented layer model. The publication provided by Joshi and colleagues (Joshi et al., 2015) describes the utilization of the Word2Vec tool for improvement purposes regarding NER in the e-commerce domain. NER is concerned with identifying entities of pre-defined categories in text. Category structures are common in e-commerce marketplaces and improve product search for users. Better categorization of items also helps with recommendation tasks. In this case the researchers focused their analysis on item titles. NER based on item titles is a hard task due to the limited information content. Item titles are short and often fuzzy and therefore more complicated to analyze than continuous large text corpora. Entity features were human annotated which classifies this research as supervised. Word Embeddings can contribute as additional features for improving NER tasks. The authors trained Word Embeddings with the Word2Vec tool. A comparison of the cbow and skip-gram models showed a better applicability of the skip-gram model for their specific purpose. As a NER classifier a Conditional Random Field (CRF) model was used. The results showed promising improvements of their task leveraging Word Embeddings. Additionally tested were in vs. out-domain data for training of the Word Embeddings. In-domain data showed more useful than out-domain data but a combination of both showed the best results for this specific task. This fact is an indicator for the usefulness of the Word2Vec tool for specific tasks, trained with specific data.

Evaluation of a selected Use Case on the basis of the layer model

Word2Vec was utilized for the computation of word vectors relating to Layer 0. A CRF model was used for classification concerning Layer 1. NER is the focus of the publication and is classified as a basic task on Layer 2. NER for e-commerce supports and improves recommender systems as stated in their publication. Recommender systems are located on Layer 3. E-commerce was classified to the Enterprise domain in this thesis, relating to Layer 4 of the presented model.

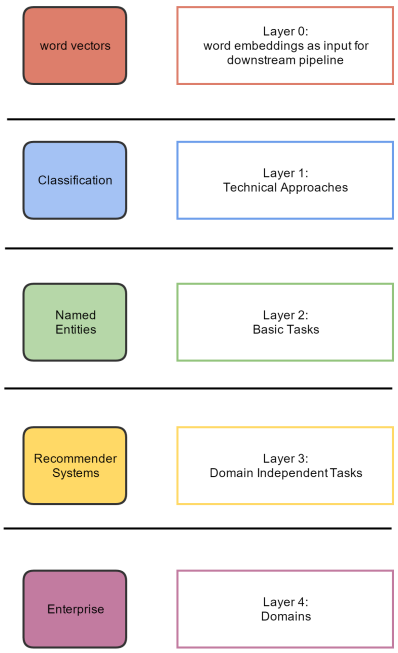


Figure 4.5.: A selected Use Case evaluated on the basis of the layer model. The figure displays only the Use Case specific selection regarding the layer content

Part III.

Evaluation of the Literature

5. Statistical evaluation of the literature

The following paragraphs describe the statistical evaluation of the literature snapshot.

The prime interest of the evaluation based on the literature snapshot, was a distribution of the literature across domains. The evaluation is based on the 989 publications extracted from the citation list as outlined in the chapter Methodology.

Additional to the extracted and in Section 4.4 described domains, the corpus contains literature impossible to evaluate. The literature consists of published research in a language other than English. Therefore these publications were classified to the category 'Other' as displayed in the following two graphs.
The following graph displays the number of publications distributed across domains.

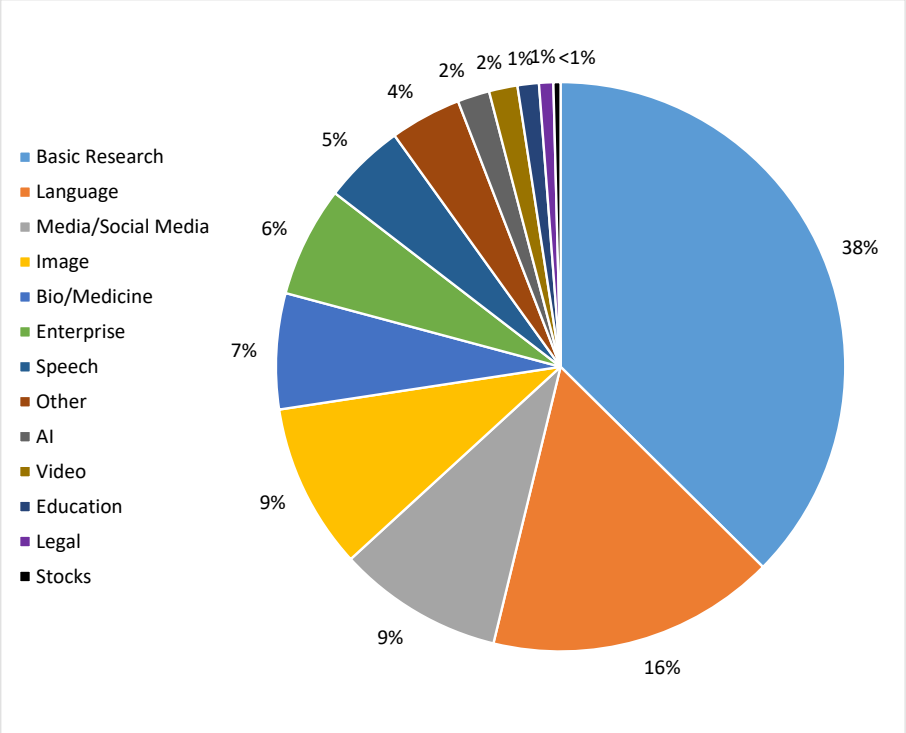


Figure 5.1.: Statistical distribution of publications over domains in the literature snapshot sorted by decreasing percentage.

The ensuing graph shows the statistical distribution of publications across domains in descending order. The evaluation is identical as in the presented graph before but for illustrative reasons displayed in a column chart sorted by decreasing percentage. Again the base is 989 publications. The horizontal axis of this chart shows the different domains extracted from the snapshot as described in Section 4.4. The vertical axis shows the percentage of the respective domain respective to the corpus.

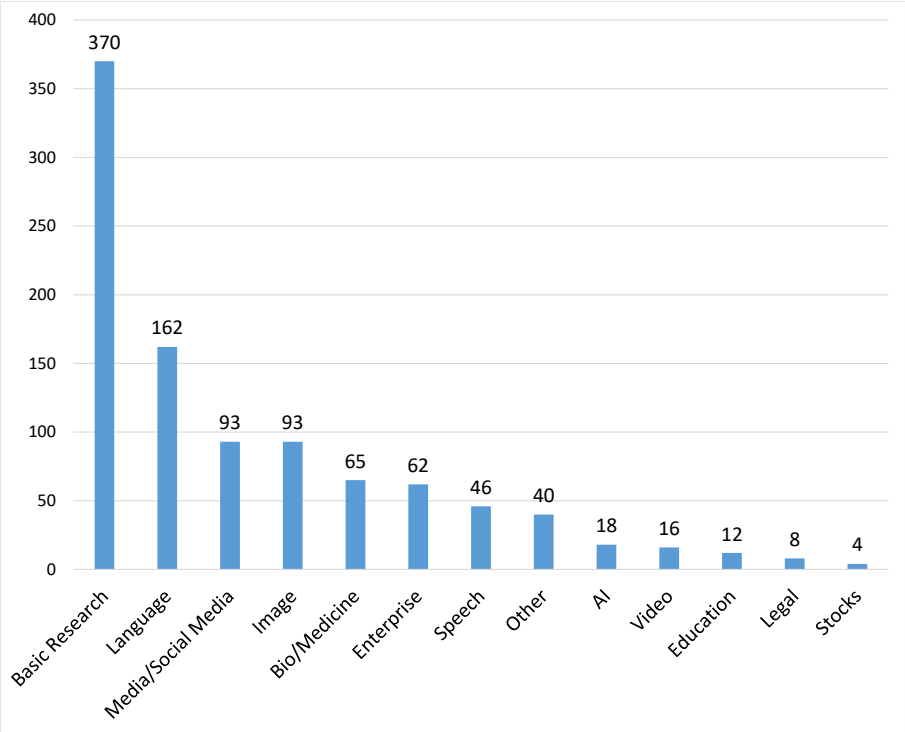


Figure 5.2.: Statistical distribution of publications across domains from the literature snapshot in descending order by number of publications

The data supports the subjective perception that an increasing part of the published literature concerns domain independent research. Despite literature researching Word Embeddings solely for improvement purposes on various NLP tasks and technical approaches, the corpus also includes literature just citing the work by Mikolov et al. for distinctive purposes to their own research but not actually applying any sort of Word Embeddings. Due to the research focus of this thesis, these publications were neither filtered nor excluded from the corpus. A further analysis regarding this fact is planned as one of the next steps for future literature evaluation.

To conclude this chapter, the following tables display the 30 most cited articles referencing the publication of Mikolov and colleagues (Mikolov et al., 2013a) and the evolvement of their own citations from 09/2016-03/2017. A list of most cited articles in a research field is a common strategy to measure the importance and impact of an article (Martín-Martín et al., 2016). The purpose of this lists is to give a fuller picture of the development speed on the topic of word embeddings and a potential starting point for future research. The first table presents the list extracted on September 01, 2016 and the second table the list extracted on March 01, 2017. Both tables have shortened titles for illustration purposes.

A comparison of the two snapshots illustrates a major overall increase of citation counts on nearly every publication. Deeper inspection revealed, that a major part of the list are publications trying to explain the work of Mikolov and colleagues. Another finding is the little substitution and interchange of the literature. The overall development of the list leads to the assumption that the uptrend in the research field is quite recognizable but rather few recent publications referencing Mikolov et al.'s work scored in the top ranks. Future work comparing the list with an overall most cited list in the research field of Word Embeddings will provide further insights on these trends.

Table 5.1.: List of 30 most cites publications referencing to (Mikolov et al., 2013a) extracted from Google Scholar on September 01, 2016

	title	authors	cites
1	Distributed representations of	(Mikolov et al., 2013c)	2271
2	Imagenet large scale visual re	(Russakovsky et al., 2015)	1268
3	Glove: Global Vectors for Word	(Pennington et al., 2014)	899
4	Distributed Representations of	(Le and Mikolov, 2014)	631
5	Show and tell: A neural image	(Vinyals et al., 2015b)	500
6	Deep Learning: Methods and App	(Deng et al., 2014)	303
7	Knowledge vault: A web-scale a	(Dong et al., 2014)	270
8	Devise: A deep visual-semantic	(Frome et al., 2013)	264
9	Intriguing properties of neura	(Szegedy et al., 2013)	236
10	Neural word embedding as impli	(Levy and Goldberg, 2014b)	210
11	Exploiting similarities among	(Mikolov et al., 2013b)	206
12	Unifying visual-semantic embed	(Kiros et al., 2014b)	174
13	Dependency-Based Word Embeddin	(Levy and Goldberg, 2014a)	169
14	Improving distributional simil	(Levy et al., 2015)	151
15	Simlex-999: Evaluating semanti	(Hill et al., 2016)	135
16	Bilingual Word Embeddings for	(Zou et al., 2013)	133
17	Learning word embeddings effic	(Mnih and Kavukcuoglu, 2013)	133
18	Multimodal Neural Language Mod	(Kiros et al., 2014a)	130
19	Linguistic Regularities in Spa	(Levy et al., 2014)	126
20	How to construct deep recurren	(Pascanu et al., 2013)	123
21	Tailoring Continuous Word Repr	(Bansal et al., 2014)	121
22	Skip-thought vectors	(Kiros et al., 2015)	116
23	Grammar as a foreign language	(Vinyals et al., 2015a)	110
24	word2vec Explained: deriving M	(Goldberg and Levy, 2014)	99
25	Deepwalk: Online learning of s	(Perozzi et al., 2014)	97
26	Convolutional neural network a	(Hu et al., 2014)	94
27	A Neural Network for Factoid Q	(Iyyer et al., 2014b)	92
28	Explain images with multimodal	(Mao et al., 2014)	90
29	Deep learning	(Goodfellow et al., 2016)	89
30	Deep Convolutional Neural Netw	(Dos Santos and Gatti, 2014)	88

Table 5.2.: List of 30 most cites publications referencing to (Mikolov et al., 2013a) extracted from Google on Scholar March 01, 2017

	title	author	cites
1	Distributed representations of	(Mikolov et al., 2013c)	3195
2	Imagenet large scale visual re	(Russakovsky et al., 2015)	1969
3	Glove: Global Vectors for Word	(Pennington et al., 2014)	1322
4	Distributed Representations of	(Le and Mikolov, 2014)	903
5	Show and tell: A neural image	(Vinyals et al., 2015b)	686
6	Tensorflow: Large-scale machin	(Abadi et al., 2016)	579
7	Deep Learning: Methods and App	(Deng et al., 2014)	438
8	Devise: A deep visual-semantic	(Frome et al., 2013)	340
9	Knowledge vault: A web-scale a	(Dong et al., 2014)	323
10	Intriguing properties of neura	(Szegedy et al., 2013)	330
11	Neural word embedding as impli	(Levy and Goldberg, 2014b)	268
12	Exploiting similarities among	(Mikolov et al., 2013b)	249
13	Vqa: Visual question answering	(Antol et al., 2015)	224
14	Dependency-Based Word Embeddin	(Levy and Goldberg, 2014a)	219
15	Unifying visual-semantic embed	(Kiros et al., 2014b)	217
16	Improving distributional simil	(Levy et al., 2015)	211
17	Deep learning	(Deng et al., 2014)	237
18	Skip-thought vectors	(Kiros et al., 2015)	202
19	Grammar as a foreign language	(Vinyals et al., 2015a)	171
20	Simlex-999: Evaluating semanti	(Hill et al., 2016)	176
21	Bilingual Word Embeddings for	(Zou et al., 2013)	165
22	Deepwalk: Online learning of s	(Perozzi et al., 2014)	175
23	Learning word embeddings effic	(Mnih and Kavukcuoglu, 2013)	155
24	How to construct deep recurren	(Pascanu et al., 2013)	156
25	Multimodal Neural Language Mod	(Kiros et al., 2014a)	152
26	Linguistic Regularities in Spa	(Levy et al., 2014)	154
27	Mind's eye: A recurrent visual	(Chen and Lawrence Zitnick, 2015)	147
28	word2vec Explained: deriving M	(Goldberg and Levy, 2014)	150
29	Convolutional neural network a	(Hu et al., 2014)	142
30	Character-aware neural languag	(Kim et al., 2015)	142

6. Further Developments

This section provides a short overview of developments on the word2vec topic. It includes selected examples of methods with alternative computation of word vectors, progressions on methods for utilization of Word Embeddings on text, as well as embedding methods for non-text applications. This section refers to research question **Q5.** *What further technological developments of Word2Vec can be identified in the body of literature?* and pictures the importance of the Word2Vec method created by Mikolov and colleagues and the speed of development in the field of Word Embeddings.

Alternative computation of word vectors.
Inspired by the success of the methods created by Mikolov et al., Pennington and colleagues (Pennington et al., 2014) created a method leveraging co-occurrence matrices, matrix factorization and a local window method explicitly not relying on a neural network. The created method named GloVe outperformed Word2Vec on several tasks and displays a promising alternative for word vector creation. A comparison of the two methods can be found in (Shi and Liu, 2014).
Other Word Embedding methods like Gaussian Embeddings published by Vilnis and McCallum (Vilnis and McCallum, 2014), reinvented count based models like (Lebret and Collobert, 2015) and character based embedding models like (Ling et al., 2015b) are intriguing research contributions in the field but did not nearly catch comparable attention as Word2Vec or GloVe.

Progression of the Word2Vec method and continuous word vector models on textual tasks or applications
This paragraph shortly outlines selected developments displayed in the following figure[1]:

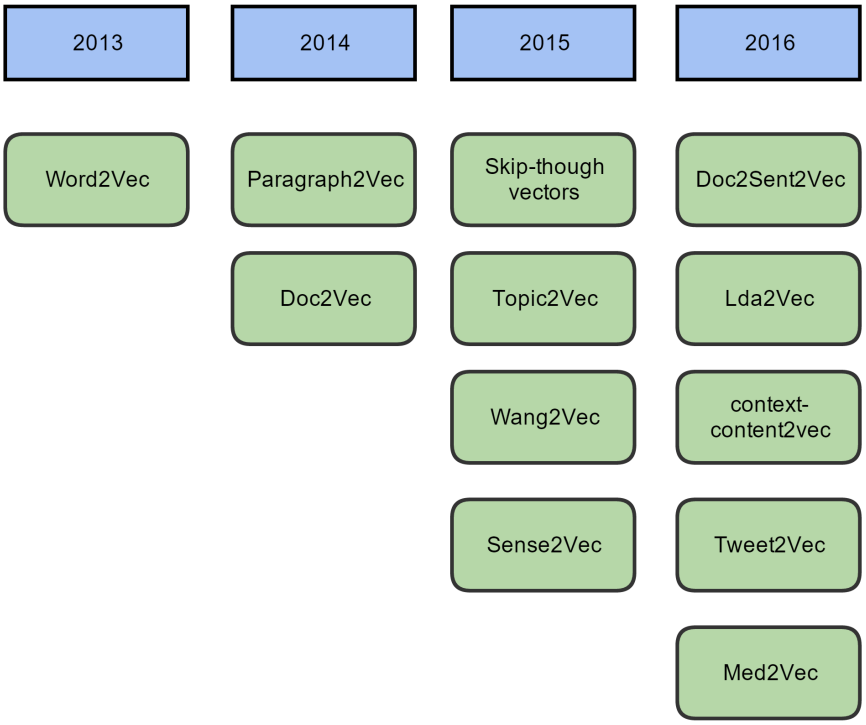


Figure 6.1.: Overview of further developments on textual tasks or applications sorted by year from 2013-2016

Evolving from word vectors and following the work of (Mikolov et al., 2013a) and (Mikolov et al., 2013c), Le and Mikolov created in 2014 the so called methods "Paragraph Vector" (Le and Mikolov, 2014), which later became known as Paragraph2Vec in the majority of the literature. The Paragraph Vector algorithm "learns fixed-length feature representations from variable-length pieces of texts, such as sentences, paragraphs, and documents" (Le and Mikolov, 2014). Řehůřek and Sojka (Řehůřek and Sojka, 2010) developed a well established Python library named Gensim for NLP tasks which also includes the Doc2Vec method based on Paragraph Vectors. Doc2Vec is widely used in the body of literature and many adaptions and implementations exist.
A very interesting list of the Gensim library adopters can be found on (Řehůřek, 2011), which shows the significant acceptance of the framework beyond the research community.

[1]The figure includes the Word2Vec tool for a better illustration but it is not described explicitly in this section.

From the literature of 2015 especially following adaptions and progressions of Word2Vec are interesting:
Kiros et al. created skip-though vectors (Kiros et al., 2015), a method which embeds whole sentences instead of concatenating word vectors.
Niu and Dai proposed Topic2Vec (Niu et al., 2015), a algorithm for topic modeling using Word2Vec as well as topic embeddings in the same vector space. With their method they improved topic modeling in contrast to LDA implementations (Niu et al., 2015).
Improvements on syntactic tasks were made by Ling et al. (Ling et al., 2015a) by slightly modifying Word2Vec to involve word ordering, mainly for POS tagging and dependency parsing. The authors created an implementation called Wang2Vec (Ling et al., 2015a) which despite its usefulness to the mentioned tasks found rather little utilization in the existing body of literature.
Trask and colleagues improved Word2Vec on Word Sense Disambiguation by applying Part-of-Speech tagging, before training Word Embeddings. They named their method Sense2Vec (Trask et al., 2015).

In 2016, several interesting progressions were published in the field of Word Embeddings.
Ganesh et al. (Gupta et al., 2016a) developed a method they called Doc2Send2Vec which embeds documents on a sentence vector base which is a slight twist, compared to the sole concatenation of word vectors Le and Mikolov (Le and Mikolov, 2014) did. They tested their method on different document classification tasks and achieved promising results (Gupta et al., 2016a).
With a combination of LDA and Word2Vec Moody (Moody, 2016) created a method called Lda2Vec which leverages both techniques for topic modeling on document level.
Grbovic and colleagues designed Context2Vec and Content2Vec which resulted in an overall Context-Content2Vec model (Grbovic et al., 2015a). Its purpose is the classification and expansion of search queries for better online advertising (Grbovic et al., 2015a).
Tweet2Vec was created by Dhingra and colleagues (Dhingra et al., 2016). It utilizes Word2Vec for hashtag prediction based on character and word embeddings for complete tweet embeddings (Dhingra et al., 2016).
Beyond the usual domains Choi et al. (Choi et al., 2016) developed Med2Vec, a method for medical concept classification based on Word2Vec.

Development to non-text applications with word embeddings related to Word2Vec. The following paragraph shortly outlines selected developments displayed in the following figure[2]:

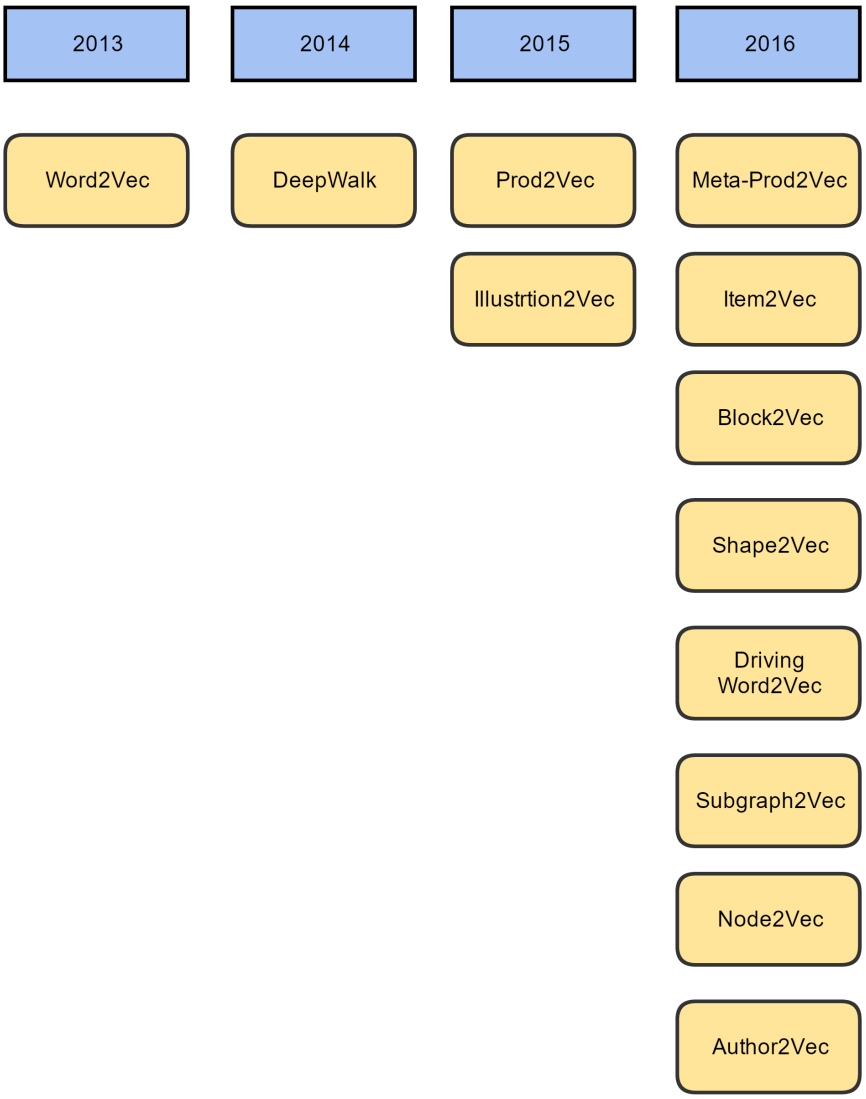


Figure 6.2.: Overview of further developments on **non-textual** tasks or applications sorted by year from 2013-2016

[2]The figure includes the Word2Vec tool for a better illustration but it is not described explicitly in this section.

Although non-text developments with Word Embeddings had a slow start in the research community they picket up the pace throughout recent years.
The most interesting contribution in 2014 was published by Perozzi and colleagues (Perozzi et al., 2014). They developed a graph based embedding method named DeepWalk relying on Word2Vec for social network embeddings (Perozzi et al., 2014).

Two developments are recognizable emerging from 2015.
Grbovic and colleagues proposed Prod2Vec (Grbovic et al., 2015b), a method for product recommendation for e-commerce via embedding of products treated as words and purchase sequences as sentences.
Illustrtion2vec (Saito and Matsui, 2015) was created for similar image retrieval from image databases by creating feature vectors for the respective illustrations.
In 2016 more and more researchers tried to exploit the uptrending methods, resulting in several publications.
Based on the Prod2Vec work from Grbovic and colleagues (Grbovic et al., 2015b), Vasile et al. created a method called Meta-Prod2Vec (Vasile et al., 2016) refining the original model by taking product meta data into account which led to significant improvements in the quality of product recommendations.
Barkan and Koenigstein (Barkan and Koenigstein, 2016) proposed a method called Item2Vec which uses Word2Vec for item-based Collaborative Filering (CF) for product recommendations.
Block2Vec created by Dai et al. (Dai et al., 2016) aims for block correlation mining in storage systems. They developed a method based on the Word2Vec idea specifically tailored for their task and reached comparable results with well established methods in the field.
From the image domain Tasse and Dogson leveraged the Word2Vec tool for and overall shape descriptor for image description and called it shape2vec (Tasse and Dodgson, 2016).
Fuchida and colleagues studied Word2Vec in the context of driving data in the form of images and videos. They called their method Driving Word2Vec (Fuchida et al., 2016) and intriguingly compared the properties of natural driving behaviour with semantic relationships on natural language.
Another graph based embedding method was created by Narayanan et al. (Narayanan et al., 2016) combining Word Embeddings and established graph algorithms. The developed model named Subgraph2Vec outperformed established graph methods and shows promising results for future research in the respective field.
Nearly parallel Grover and Leskovec also researched and published a method called Node2Vec developed for network environments based on Word Embeddings (Grover and Leskovec, 2016).
The Author2Vec model (Ganguly et al., 2016) embeds citation networks for authors reflecting their published profile. Leveraging the idea of DeepWalk the article describes a graph based method implementing the authors citational network in low dimensional space. Possible applications for this method would be link prediction or potential co-author identification.

The development of the technology beyond text shows its applicability. Nearly all of the examined publications presented above, were extracted from the literature snapshot which illustrates the recognition and evolvement of the topic.

7. Lessons Learned

This chapter presents the main findings recognized throughout the development process of this thesis.

The distribution over domains displayed in chapter 5 illustrates several tendencies. Mainly, over a third of the examined research was classified as basic research relating to research with no directly linked purpose for a specific domain. The data appears to suggest that a huge proportion of research conducted in the field of Word Embeddings tries to leverage or improve Word Embeddings for technical approaches, basic tasks or domain independent tasks. Correlated with the increasing trend of further developments on non-textual tasks or applications from Chapter 6, the main understanding is that the breakthrough of Mikolov and colleagues led to an elevated utilization of Word Embeddings for different NLP and non-NLP tasks. Especially the trend for publications regarding research applied to non-text tasks illustrates the transferability of the idea of Word Embeddings, which indicates promising and interesting directions for the future.

A second point extracted from the domain evaluation is that Word Embeddings are applied to and mostly tried out in a broad range of different domains. This underlines the applicability of the idea of Word Embeddings. But, since the amount of textual data increases steadily in nearly every domain, also the need of qualitative applications for NLP tasks arises.

In contrast to the huge amount of research conducted domain independently, some real world applications (Řehůřek, 2011) adopted Word Embeddings. The reference provides a list of organizations using the Gensim library for various NLP tasks. The library includes the Word2Vec and other Word Embedding algorithms and illustrates the tendency that they are more than a research topic and enrich NLP and non-NLP software alike.

Another finding was that continuous word vectors so far are mostly created with domain or even with task specific text input. This creates problems with transferability of the research results to other tasks or applications. Task specific embeddings create higher quality of the task results (Levy and Goldberg, 2014a) which is alright for basic research but not for continuously improving and potentially cross-domain adapting real world applications. A few publications mention and tackle this problem but to utilize Word Embeddings to the full extend this is one hurdle to overcome.

A huge part of the literature applies Word Embeddings to application pipelines in which downstream tasks are implemented supervised. To exploit the advantageous unsupervised properties of methods like Word2Vec in more depth, several downstream tasks leveraging word vectors have to be researched for unsupervised implementations for fully automated and cost efficient systems.

One of the more illuminating discovery during the development of this thesis was, that lots of research combines several technical approaches, basic or DI tasks to realize applications. The advantages of these attempts are mostly novel approaches in their research domain, black box alike implementations of tasks, API's and mini web services[1]. These implementations and services also increase the possibilities to easier combine several methods and explore the applicability for the own research. This also provides easier access to the field for beginners but on the other hand a major drawback is that analysis of the state-of-the art publications gain a lot more complexity. This also is a validating argument for the layer model presented in this thesis providing easier overview and understanding for the majority of used tasks. Despite, it is surely is a great motivation for future improvements on the model following in the next chapter.

[1] http://blog.algorithmia.com/cloud-hosted-deep-learning-models/

Part IV.

Future Work and Conclusion

8. Future Work

The layer model introduced in this thesis provides a natural guide to future research. Much research remains to be done to evolve and improve the presented model. Extension of the descriptions regarding the different layers for introductory purposes and improving the functionality and usability of the layer model are the next steps towards an easier and more qualitative analyzing tool.
The basis for evolvements of the model are further analyses of the state-of-the-art literature. In the following, several aspects are outlined portraying further research. Extending the literature analysis to domains outside of the thesis focus for a better overall view in the research field. Regarding the statistical analysis displayed in Chapter 5, a next step will be the separation of literature not researching or applying Word Embeddings, but nevertheless citing the original work, from the examined corpus. Additional analyses of the distribution regarding technical approaches, basic and domain independent tasks across domains will be conducted and are expected to enhance the layer model and contribute supplementary insights on the research field.
Further analysis of the 30 most cited list displayed in Chapter 5 and an overall most cited publications list in the whole field of word embeddings. This could be helpful for an analysis of the overall coherences and utilization of Word Embeddings.
Another worthwhile tasks is an overview of basic and DI tasks applying Word Embeddings which are superior to older methods in the respective field or domain. This overview would provide entry points for adopters of the technology in their respective field.

9. Conclusion

This thesis was motivated by the increasing hype concerning Word Embeddings in the recent years and to clarify its meaning. The approach of the process was twofold. A classification and evaluation of the literature referencing on Mikolov et al.'s work and to develop and to introduce a conceptual layer model for introductory purposes, regarding the research field of Word Embeddings. Setting Word Embeddings and their utilization in context, the question of a categorical overview for Use Cases with Word Embeddings arose. The first step was a classification of the literature corpus regarding domains. The distribution revealed that a major part of the research is domain independent and concerns Word Embeddings for technical approaches, basic tasks or domain independent tasks relating to the layer 1 through 3 of the later developed layer model. Further developments on the popular Word2Vec tool revealed an intriguing trend to non-textual applications leveraging the idea of Word Embeddings. Limiting factors for this thesis were on the one hand the vast amount of literature regarding Word Embeddings and on the other hand the significant differences in complexity throughout the various publications. Despite the vast amount of literature in this field, there does not exist and individual publication introducing the tasks and topics regarding Word Embeddings in an overall manner so far. The introduced layer model includes introductions to technical approaches, basic tasks, domain independent tasks and domains Word Embeddings are researched in. The layer model illustrates an overview for beginners in the field and has the potential to build a conceptual starting point for the widespread research field.

List of Figures

List of Tables

Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D. (2015). Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433.

Arora, S., Ge, R., and Moitra, A. (2012). Learning topic models–going beyond svd. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 1–10. IEEE.

Atefeh, F. and Khreich, W. (2015). A survey of techniques for event detection in twitter. *Computational Intelligence*, 31(1):132–164.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction from the web. In *IJCAI*, volume 7, pages 2670–2676.

Bansal, M., Gimpel, K., and Livescu, K. (2014). Tailoring continuous word representations for dependency parsing. In *ACL (2)*, pages 809–815.

Barkan, O. and Koenigstein, N. (2016). Item2vec: neural item embedding for collaborative filtering. In *Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on*, pages 1–6. IEEE.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL (1)*, pages 238–247.

Bartunov, S., Kondrashkin, D., Osokin, A., and Vetrov, D. (2015). Breaking sticks and ambiguities with adaptive skip-gram. *arXiv preprint arXiv:1502.07257*, pages 47–54.

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.

Boltuzic, F. and Šnajder, J. (2014). Back up your stance: Recognizing arguments in online discussions. In *Proceedings of the First Workshop on Argumentation Mining*, pages 49–58. Citeseer.

Bordes, A., Weston, J., Collobert, R., and Bengio, Y. (2011). Learning structured embeddings of knowledge bases. In *Conference on artificial intelligence*, number EPFL-CONF-192344.

Büttcher, S., Clarke, C. L., and Cormack, G. V. (2016). *Information retrieval: Implementing and evaluating search engines*. Mit Press.

Cai, F. and de Rijke, M. (2016). Learning from homologous queries and semantically related terms for query auto completion. *Information Processing & Management*, 52(4):628–643.

Chen, X. and Lawrence Zitnick, C. (2015). Mind's eye: A recurrent visual representation for image caption generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2422–2431.

Choi, E., Bahadori, M. T., Searles, E., Coffey, C., Thompson, M., Bost, J., Tejedor-Sojo, J., and Sun, J. (2016). Multi-layer representation learning for medical concepts. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1495–1504. ACM.

Chopra, A., Prashar, A., and Sain, C. (2013). Natural language processing. *Int. J. Technol. Enhanc. Emerg. Eng. Res*, 1(4):131–134.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM.

Cresci, S., Tesconi, M., Cimino, A., and Dell'Orletta, F. (2015). A linguistically-driven approach to cross-event damage assessment of natural disasters from social media messages. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1195–1200. ACM.

Dai, D., Bao, F. S., Zhou, J., and Chen, Y. (2016). Block2vec: A deep learning strategy on mining block correlations in storage systems. In *Parallel Processing Workshops (ICPPW), 2016 45th International Conference on*, pages 230–239. IEEE.

Das, R., Zaheer, M., and Dyer, C. (2015). Gaussian lda for topic models with word embeddings. In *ACL (1)*, pages 795–804.

Deng, L., Yu, D., et al. (2014). Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387.

Dhingra, B., Zhou, Z., Fitzpatrick, D., Muehl, M., and Cohen, W. W. (2016). Tweet2vec: Character-based distributed representations for social media. *arXiv preprint arXiv:1605.03481*.

Ding, X., Zhang, Y., Liu, T., and Duan, J. (2015). Deep learning for event-driven stock prediction. In *IJCAI*, pages 2327–2333.

Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. ACM.

Dos Santos, C. N. and Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. In *COLING*, pages 69–78.

Edmonds, P. and Agirre, E. (2008). Word sense disambiguation. *Scholarpedia*, 3(7):4358. revision ♯90370.

Fantechi, A., Gnesi, S., Lami, G., and Maccari, A. (2003). Applications of linguistic techniques for use case analysis. *Requirements Engineering*, 8(3):161–170.

Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89.

Ferrero, J., Agnès, F., Besacier, L., and Schwab, D. (2017). Using word embedding for cross-language plagiarism detection. In *European Association for Computational Linguistics (EACL)*.

Fonseca, E. R., Rosa, J. L. G., and Aluísio, S. M. (2015). Evaluating word embeddings and a revised corpus for part-of-speech tagging in portuguese. *Journal of the Brazilian Computer Society*, 21(1):2.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al. (2013). Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *ACL (1)*, pages 1199–1209.

Fuchida, Y., Taniguchi, T., Takano, T., Mori, T., Takenaka, K., and Bando, T. (2016). Driving word2vec: Distributed semantic vector representation for symbolized naturalistic driving data. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 1313–1320. IEEE.

Ganguly, S., Gupta, M., Varma, V., Pudi, V., et al. (2016). Author2vec: Learning author representations by combining content and link information. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 49–50.

Goldberg, Y. (2016). A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420.

Goldberg, Y. and Levy, O. (2014). word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT Press.

Grbovic, M., Djuric, N., Radosavljevic, V., Silvestri, F., and Bhamidipati, N. (2015a). Context-and content-aware embeddings for query rewriting in sponsored search. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 383–392. ACM.

Grbovic, M., Radosavljevic, V., Djuric, N., Bhamidipati, N., Savla, J., Bhagwan, V., and Sharp, D. (2015b). E-commerce in your inbox: Product recommendations at scale. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1809–1818. ACM.

Grover, A. and Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM.

Guha, S., Joshi, A., and Varma, V. (2015). Siel: Aspect based sentiment analysis in reviews. *SemEval-2015*, 759.

Gupta, M., Varma, V., et al. (2016a). Doc2sent2vec: A novel two-phase approach for learning document representation. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 809–812. ACM.

Gupta, N., Podder, S., Annervaz, K., and Sengupta, S. (2016b). Domain ontology induction using word embeddings. In *Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on*, pages 115–119. IEEE.

Habibi, M. and Popescu-Belis, A. (2015). Keyword extraction and clustering for document recommendation in conversations. *IEEE/ACM Transactions on audio, speech, and language processing*, 23(4):746–759.

Hill, F., Reichart, R., and Korhonen, A. (2016). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Hu, B., Lu, Z., Li, H., and Chen, Q. (2014). Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*, pages 2042–2050.

Iyyer, M., Boyd-Graber, J., and Daumé III, H. (2014a). Generating sentences from semantic vector space representations. In *Proc. NIPS Workshop on Learning Semantics*.

Iyyer, M., Boyd-Graber, J. L., Claudino, L. M. B., Socher, R., and Daumé III, H. (2014b). A neural network for factoid question answering over paragraphs. In *EMNLP*, pages 633–644.

Iyyer, M., Enns, P., Boyd-Graber, J., and Resnik, P. (2014c). Political ideology detection using recursive neural networks. In *Proceedings of the Association for Computational Linguistics*, pages 1113–1122.

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666.

Joshi, M., Hart, E., Vogel, M., and Ruvini, J.-D. (2015). Distributed word representations improve ner for e-commerce. In *Proceedings of NAACL-HLT*, pages 160–167.

Kågebäck, M., Mogren, O., Tahmasebi, N., and Dubhashi, D. (2014). Extractive summarization using continuous vector space models. In *Proceedings of the 2nd Workshop on*

Continuous Vector Space Models and their Compositionality (CVSC)@ EACL, pages 31–39. Citeseer.

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. M. (2015). Character-aware neural language models. *arXiv preprint arXiv:1508.06615*.

Kiros, R., Salakhutdinov, R., and Zemel, R. S. (2014a). Multimodal neural language models. In *Icml*, volume 14, pages 595–603.

Kiros, R., Salakhutdinov, R., and Zemel, R. S. (2014b). Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Kolluru, S. and Mukherjee, P. (2016). Query clustering using segment specific context embeddings. *arXiv preprint arXiv:1608.01247*.

Le, Q. V. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196.

Lebret, R. and Collobert, R. (2015). Rehabilitation of count-based models for word vector representations. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 417–429. Springer.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al. (2015). Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Levy, O. and Goldberg, Y. (2014a). Dependency-based word embeddings. In *ACL (2)*, pages 302–308. Citeseer.

Levy, O. and Goldberg, Y. (2014b). Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*, pages 2177–2185.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Levy, O., Goldberg, Y., and Ramat-Gan, I. (2014). Linguistic regularities in sparse and explicit word representations. In *CoNLL*, pages 171–180.

Li, Q., Shah, S., Liu, X., Nourbakhsh, A., and Fang, R. (2016a). Tweet topic classification using distributed language representations. In *Web Intelligence (WI), 2016 IEEE/WIC/ACM International Conference on*, pages 81–88. IEEE.

Li, S., Chua, T.-S., Zhu, J., and Miao, C. (2016b). Generative topic embedding: a continuous representation of documents. In *Proceedings of The 54th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Li, Y., Xu, L., Tian, F., Jiang, L., Zhong, X., and Chen, E. (2015). Word embedding revisited: A new representation learning and explicit matrix factorization perspective. In *IJCAI*, pages 3650–3656.

Ling, W., Dyer, C., Black, A. W., and Trancoso, I. (2015a). Two/too simple adaptations of word2vec for syntax problems. In *HLT-NAACL*, pages 1299–1304.

Ling, W., Luís, T., Marujo, L., Astudillo, R. F., Amir, S., Dyer, C., Black, A. W., and Trancoso, I. (2015b). Finding function in form: Compositional character models for open vocabulary word representation. *arXiv preprint arXiv:1508.02096*.

Liu, P., Joty, S. R., and Meng, H. M. (2015). Fine-grained opinion mining with recurrent neural networks and word embeddings. In *EMNLP*, pages 1433–1443.

Ma, B., Yuan, H., Wan, Y., Qian, Y., Zhang, N., and Ye, Q. (2016). Public opinion analysis based on probabilistic topic modeling and deep learning. *PACIS 2016 Proceedings*.

Madnani, N. and Dorr, B. J. (2010). Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3):341–387.

Manning, C. D., Raghavan, P., Schütze, H., et al. (2008). *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.

Mao, J., Xu, W., Yang, Y., Wang, J., and Yuille, A. L. (2014). Explain images with multimodal recurrent neural networks. *arXiv preprint arXiv:1410.1090*.

Martín-Martín, A., Orduña-Malea, E., Ayllon, J. M., and López-Cózar, E. D. (2016). The counting house: measuring those who count. presence of bibliometrics, scientometrics, informetrics, webometrics and altmetrics in the google scholar citations, researcherid, researchgate, mendeley & twitter. *arXiv preprint arXiv:1602.02412*.

Maynard, D., Bontcheva, K., and Augenstein, I. (2016). Natural language processing for the semantic web. *Synthesis Lectures on the Semantic Web: Theory and Technology*, 6(2):1–194.

McCormik, C. (2016). Word2vec tutorial - the skip-gram model. Last accessed on Feb 28, 2017.

Meho, L. I. and Yang, K. (2007). Impact of data sources on citation counts and rankings of lis faculty: Web of science versus scopus and google scholar. *Journal of the american society for information science and technology*, 58(13):2105–2125.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Le, Q. V., and Sutskever, I. (2013b). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013c). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013d). Linguistic regularities in continuous space word representations. In *Hlt-naacl*, volume 13, pages 746–751.

Mitchell, J. and Lapata, M. (2010). Composition in distributional models of semantics. *Cognitive science*, 34(8):1388–1429.

Mnih, A. and Kavukcuoglu, K. (2013). Learning word embeddings efficiently with noise-contrastive estimation. In *Advances in neural information processing systems*, pages 2265–2273.

Moens, M.-F. (2013). Argumentation mining: Where are we now, where do we want to be and how do we get there? In *Post-Proceedings of the 4th and 5th Workshops of the Forum for Information Retrieval Evaluation*, page 2. ACM.

Mohammad, S. M., Kiritchenko, S., Sobhani, P., Zhu, X., and Cherry, C. (2016). Semeval-2016 task 6: Detecting stance in tweets. *Proceedings of SemEval*, 16.

Moody, C. E. (2016). Mixing dirichlet topic models and word embeddings to make lda2vec. *arXiv preprint arXiv:1605.02019*.

Naderi, N. and Hirst, G. (2014). Argumentation mining in parliamentary discourse. In *International Workshop on Empathic Computing*, pages 16–25. Springer.

Nadkarni, P. M., Ohno-Machado, L., and Chapman, W. W. (2011). Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551.

Narayanan, A., Chandramohan, M., Chen, L., Liu, Y., and Saminathan, S. (2016). subgraph2vec: Learning distributed representations of rooted sub-graphs from large graphs. *arXiv preprint arXiv:1606.08928*.

Nenkova, A., McKeown, K., et al. (2011). Automatic summarization. *Foundations and Trends® in Information Retrieval*, 5(2–3):103–233.

Nguyen, T. H. and Grishman, R. (2015). Event detection and domain adaptation with convolutional neural networks. In *ACL (2)*, pages 365–371.

Niu, L., Dai, X., Zhang, J., and Chen, J. (2015). Topic2vec: learning distributed representations of topics. In *Asian Language Processing (IALP), 2015 International Conference on*, pages 193–196. IEEE.

Nivre, J. (2005). Dependency grammar and dependency parsing. *MSI report*, 5133(1959):1–32.

OCallaghan, D., Greene, D., Carthy, J., and Cunningham, P. (2015). An analysis of the coherence of descriptors in topic modeling. *Expert Systems with Applications*, 42(13):5645–5657.

Ozsoy, M. G. (2016). From word embeddings to item recommendation. *arXiv preprint arXiv:1601.01356*.

Pascanu, R., Gulcehre, C., Cho, K., and Bengio, Y. (2013). How to construct deep recurrent neural networks. *arXiv preprint arXiv:1312.6026*.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM.

Ravi, K. and Ravi, V. (2015). A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems*, 89:14–46.

Řehůřek, E. (2011). gensim adopters. https://github.com/RaRe-Technologies/gensim♯adopters. Last accessed on Feb 28, 2017.

Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Rodriguez, A. and Laio, A. (2014). Clustering by fast search and find of density peaks. *Science*, 344(6191):1492–1496.

Rong, X. (2014). word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*.

Roy, D., Paul, D., Mitra, M., and Garain, U. (2016). Using word embeddings for automatic query expansion. *arXiv preprint arXiv:1606.07608*.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252.

Sahlgren, M. (2008). The distributional hypothesis. *Italian Journal of Linguistics*, 20(1):33–54.

Saito, M. and Matsui, Y. (2015). Illustration2vec: a semantic vector representation of illustrations. In *SIGGRAPH Asia 2015 Technical Briefs*, page 5. ACM.

Shi, R., Wang, H., Wang, T., Hou, Y., and Tang, Y. (2016). Similarity search combining query relaxation and diversification. *arXiv preprint arXiv:1611.04689*.

Shi, T. and Liu, Z. (2014). Linking glove with word2vec. *arXiv preprint arXiv:1411.5595*.

Sobhani, P., Inkpen, D., and Matwin, S. (2015). From argumentation mining to stance classification. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 67–77.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Tasse, F. P. and Dodgson, N. (2016). Shape2vec: semantic-based descriptors for 3d shapes, sketches and images. *ACM Transactions on Graphics (TOG)*, 35(6):208.

Tran, Q. H., Tran, V., Vu, T., Nguyen, M., and Pham, S. B. (2015). Jaist: Combining multiple features for answer selection in community question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval*, volume 15, pages 215–219.

Trask, A., Michalak, P., and Liu, J. (2015). sense2vec-a fast and accurate method for word sense disambiguation in neural word embeddings. *arXiv preprint arXiv:1511.06388*.

Vasile, F., Smirnova, E., and Conneau, A. (2016). Meta-prod2vec: Product embeddings using side-information for recommendation. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 225–232. ACM.

Vilnis, L. and McCallum, A. (2014). Word representations via gaussian embedding. *arXiv preprint arXiv:1412.6623*.

Vinyals, O., Kaiser, Ł., Koo, T., Petrov, S., Sutskever, I., and Hinton, G. (2015a). Grammar as a foreign language. In *Advances in Neural Information Processing Systems*, pages 2773–2781.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015b). Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164.

Wang, G., Sun, J., Ma, J., Xu, K., and Gu, J. (2014). Sentiment classification: The contribution of ensemble learning. *Decision support systems*, 57:77–93.

Wang, P., Guo, J., Lan, Y., Xu, J., Wan, S., and Cheng, X. (2015). Learning hierarchical representation model for nextbasket recommendation. In *Proceedings of the 38th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 403–412. ACM.

Weston, J., Bordes, A., Chopra, S., Rush, A. M., van Merriënboer, B., Joulin, A., and Mikolov, T. (2015). Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*.

Xun, G., Gopalakrishnan, V., Ma, F., Li, Y., Gao, J., and Zhang, A. (2016). Topic discovery for short texts using word embeddings. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pages 1299–1304. IEEE.

Yan, Z., Duan, N., Bao, J., Chen, P., Zhou, M., Li, Z., and Zhou, J. (2016). Docchat: an information retrieval approach for chatbot engines using unstructured documents. ACL.

Yang, H., Hu, Q., and He, L. (2015). Learning topic-oriented word embedding for query classification. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 188–198. Springer.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., and Hovy, E. (2016). Hierarchical attention networks for document classification. In *Proceedings of NAACL-HLT*, pages 1480–1489.

Yu, L., Hermann, K. M., Blunsom, P., and Pulman, S. (2014). Deep learning for answer sentence selection. *arXiv preprint arXiv:1412.1632*.

Zamani, H. and Croft, W. B. (2016). Estimating embedding vectors for queries. In *Proceedings of the 2016 ACM on International Conference on the Theory of Information Retrieval*, pages 123–132. ACM.

Zhang, Q., Kang, J., Qian, J., and Huang, X. (2014). Continuous word embeddings for detecting local text reuses at the semantic level. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 797–806. ACM.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Zhao, S., Lan, X., Liu, T., and Li, S. (2009). Application-driven statistical paraphrase generation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 834–842. Association for Computational Linguistics.

Zhao, S., Wang, Q., Massung, S., Qin, B., Liu, T., Wang, B., and Zhai, C. (2017). Constructing and embedding abstract event causality networks from text snippets. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 335–344. ACM.

Zou, W. Y., Socher, R., Cer, D. M., and Manning, C. D. (2013). Bilingual word embeddings for phrase-based machine translation. In *EMNLP*, pages 1393–1398.

Appendix A

Overview of the complete layer model including described approaches, tasks and domains from Chapter 4.

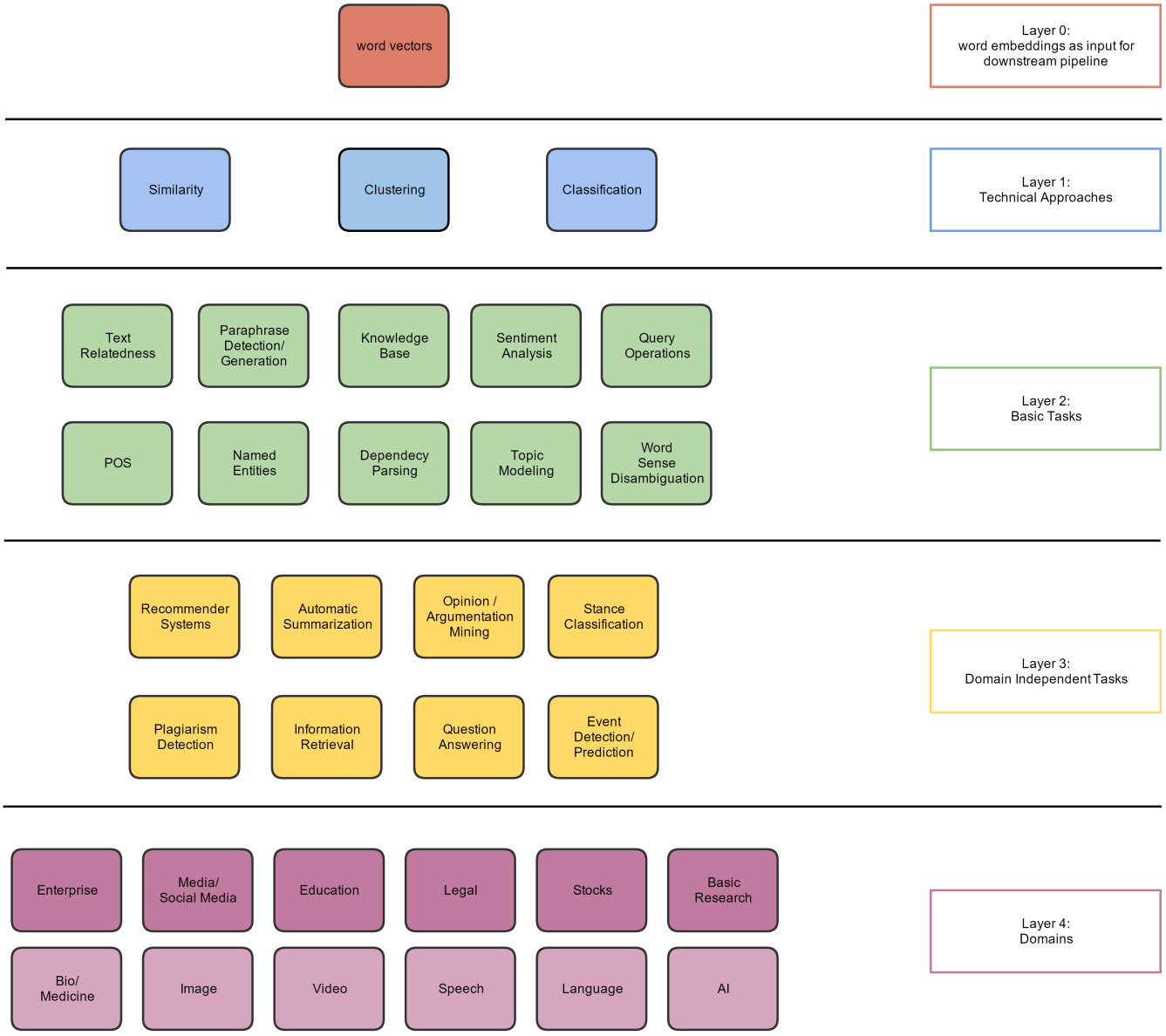


Figure 9.1.: Overview of the developed layer model including described approaches, tasks and domains

Appendix B

Screenshots of the citation counts of Mikolov et al.'s publication on Google Scholar at 1.4.16, 1.9.16, 1.11.16 and 1.3.17. 'Zitiert von: ' translates to 'Cited by: ' and is the citation index from Google Scholar.

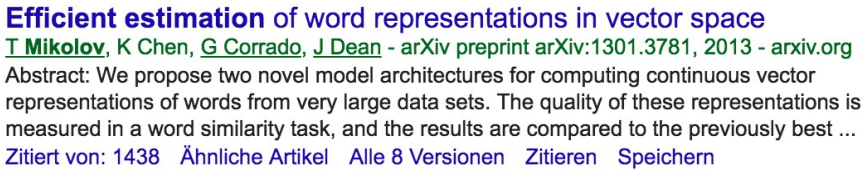


Figure 9.2.: Screenshot citation count on Mikolov et al.'s pubication on Google Scholar from 1.4.2016

Figure 9.3.: Screenshot citation count on Mikolov et al.'s pubication on Google Scholar from 1.9.2016

Figure 9.4.: Screenshot citation count on Mikolov et al.'s pubication on Google Scholar from 1.11.2016

Figure 9.5.: Screenshot citation count on Mikolov et al.'s pubication on Google Scholar from 1.3.2017