Institut für Maschinelle Sprachverarbeitung

Universität Stuttgart
Pfaffenwaldring 5b
70569 Stuttgart

Bachelorarbeit

Dimensionality and Noise in Models of Semantic Change Detection

Jens Kaiser

Studiengang: B.Sc.Informatik

Prüfer: apl. Prof. Dr. Sabine Schulte im Walde

Betreuer: Schlechtweg, Papay, Schulte im Walde

begonnen am: 01.12.2019

beendet am: 27.07.2020

Contents

1 Introduction

This thesis emerged from my work during the seminar ”Lexical Semantic Change Detection” held by Dominik Schlechtweg. During the seminar my team implemented and empirically evaluated Vector Initialisation (VI), a model proposed by Kim et al. (2014) under the assistance of Dominik Schlechtweg.[1] While Schlechtweg et al. (2019) reported inferior performance for this model compared to other approaches at the task of Semantic Change Detection, we observed a surprisingly good performance. Our implementation and choice of parameters differed significantly from Schlechtweg et al. (2019). Most prominent was the used number of dimensions. Changes to the implementation were well motivated and are further detailed in Section 3. The choice of the usual dimensionality was more of an accidental and negligent nature. Intrigued by the results of the seminar, I decided to further investigate this observation as part of my thesis.

This work has its focus on how dimensionality relates to the performance of three commonly used alignment models in Semantic Change Detection, as well as the noise these models are subjected to as a consequence of dimensionality. The three models are Vector Initialisation (VI) by Kim et al. (2014), Orthogonal Procrustes (OP) by Hamilton et al. (2016b) and Word Injection (Ferrari et al., 2017; Schlechtweg et al., 2019; Dubossarsky et al., 2019).

Yin and Shen (2018) states that optimal dimensionality is determined by noise. We concluded that VI is very susceptible to noise and has a low optimal dimensionality. This could explain why other research, (Schlechtweg et al., 2019; Shoemark et al., 2019), observed inferior performance of VI with higher dimensionality. Dubossarsky et al. (2019) show that OP captures more noise than WI, thus OP should have a lower optimal dimensionality than WI. Combining these implications we formulated four hypotheses to guide our research on this subject:

1. The optimal dimensionality is different for all three models.
2. VI has a lower optimal dimensionality than OP, and OP has a lower one the an WI.
3. VI captures more noise than OP and OP captures more noise than WI with equal dimensionality.
4. The optimal dimensionality for each model is a function of other parameters such as number of training epochs and corpus size.

As revealed in Section 5 giving an answer to the hypotheses is not straightforward as some models do not have a clear optimal dimensionality and the complexity of noise involved. Word frequency was identified as a major source of noise for VI and could be linked to a negative

[1] This work is partially published in Ahmad et al. (2020) as team ”in vain”.

impact on performance. Unfortunately, even with numerous follow-up experiments the cause for the sensitivity to frequency be explained. However, we identified several methods to reduce this influence.

2 Background

Distributional Semantics Teaching a computer any human language is no simple task. First attempts at this task for use with search engines relied on simple pattern matching. Being able to decide if two strings of text exactly match each other hardly counts as understanding a language. Many words have identical spelling or pronunciation yet have different meanings. In order to distinguish between these so-called homographs and homonyms context is needed. The distributional hypothesis postulates that words that occur in the same context tend to have similar meanings (Harris, 1954). This lays the basis of Distributional Semantics in which words are usually represented by vectors: similar words have similar vectors while the vectors of non related words differ greatly. A simplified explanation of the first implementations is as follows: Given a Corpus, a $|D| \times |D|$ matrix with all words in the corpus vocabulary D along both axes is created. Each cell holds the number of occurrences of the two respective words within a certain window. For example if word1 and word2 occur within the specified window, the value in cells (i_{word1}, j_{word2}) and (i_{word2}, j_{word1}) increases. This way, the end result is a matrix which holds a vector for each word in D. The vector of a word is characterised by its context, thus words with similar context have similar vectors. These high dimensional vectors are referred to as count vectors. While they certainly are useful in some applications, their main drawback is that information is very sparsely encoded. Meaning that a significant amount of entries in the vectors are zeros. Approaches like Singular Vector Decomposition drastically reduce dimensionality of count vectors, while retaining a majority of the information. These low dimensional vector representations of words are referred to as word embeddings.

There are many criteria for judging the quality of word embeddings. The most basic is word similarity, some word pairs are more semantically similar than others. For example the word pair (*cucumber, potato*) is more similar than (*cucumber, professor*). This similarity should be reflected in the word embeddings, where the vectors of *cucumber* and *potato* should be more similar than the vectors of *cucumber* and *professor*. Cosine similarity is often used to measure the similarity of vectors, which describes the angle between the vectors (Salton and McGill, 1983).

In this work we use Skip-gram with Negative Sampling to generate word embeddings (for a detailed description of Skip-gram see Section 4). Rather than counting the context words, SGNS creates embeddings by trying to predict the context of word. This approach yields embeddings that capture semantic similarity and relatedness even better (Baroni et al., 2014). Most state-of-

the-art models in Semantic Change Detection use SGNS to create word embeddings.

Semantic Change Detection (SCD) The goal of SCD is to detect and quantify semantic change of a variety of words. The data necessary for such a task comprises of two corpora c_1, c_2 (body's of text) from separate time periods t_1, t_2. Other multi-modal data like pictures and speech are rarely used because text is much more readily available. In some cases the data is split into more than two time periods. From here on most models based on type-based embeddings in SCD work in a similar way: (1) creating semantic word representations on c_1 and c_2, (2) aligning the word representations, and (3) measuring differences between aligned representations (Schlechtweg et al., 2019). Type-based embeddings have a vector representation for each type, i.e. word. Token-based embeddings on the other hand have a vector representation of each occurrence of a word and often require additional clustering. In this work only type-based embeddings are used. The motivation of using embeddings to detect semantic change is the following. If a word changes its meaning between two time periods, the context of the word changes as well. Thus, the embedding created on text of the first time period is different from the embedding created on the text of the second time period. Direct comparison between the two embeddings is often not possible, as the vector spaces may not be aligned. Later we will introduce three methods which solve this problem with different approaches. Once aligned, semantic changes can be detecting by comparing the two embeddings.

3 Related Work

Different models for semantic change detection are influenced by noise in varying amounts. We regard information contained within the semantic representation capturing anything but semantic relations between words as noise (e.g. word frequency). Sources of noise include the corpora, the representation method and alignment techniques. One method to dampen noise is to combine the usually small and separate training data into one larger set (Dubossarsky et al., 2019). Another method is to use already learned weights within the Neural Network as a base for incremental training, as proposed in Kaji and Kobayashi (2017) and Peng et al. (2017). Their models modified the learning process to support consecutive training by one Neural Network. This allows for word embeddings which can be updated by new data, while the quality of the embeddings is almost identical to methods where all the data is given in advance. The model proposed by Kim et al. (2014) does use incremental training to detect semantic change, but without adjustments to the learning process. Later we will show how this unmodified application leads to unwanted side effects, like a bias that reports higher change for more frequent words. This is not uncommon for models in SCD: Dubossarsky et al. (2017) show that many models are significantly influenced by

word frequency. In both Schlechtweg et al. (2019) and Shoemark et al. (2019), which compared different models for SCD on a standardised test set, the incremental model by Kim et al. (2014) was outperformed by models using post-processing alignment (OP). The implementation of VI in Schlechtweg et al. (2019) only used the word vectors for initialisation, while context vectors were initialised with random values. Their work promised a fair comparison between models by using the same hyper-parameters, like dimensionality, for all models. Yin and Shen (2018) show a connection between dimensionality of word embeddings and noise. If dimensionality is chosen too low, the created embeddings only capture some semantic relations. If chosen too high the embeddings over-fit on the training data and contain corpus specific information, which is considered to be noise. Thus, in connection with models being subjected to varying amounts of noise they may have to be compared with different dimensionalities.

4 System Overview

4.1 Skip-Gram with Negative Sampling (SGNS)

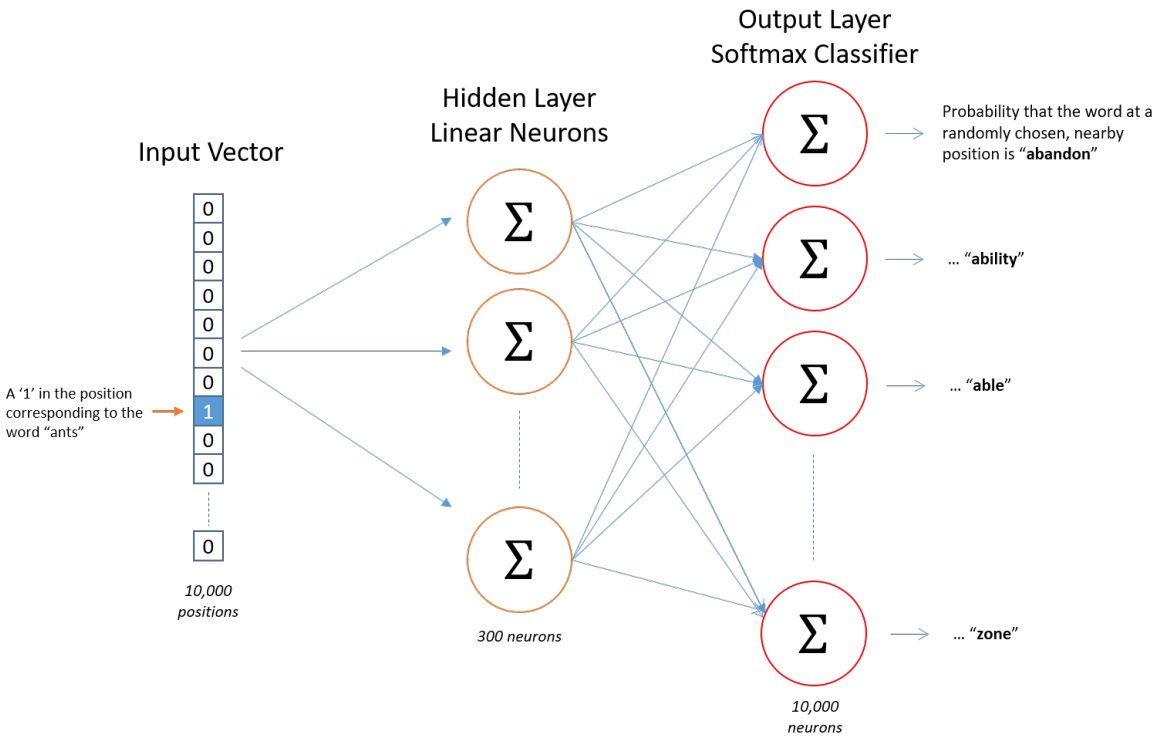


Figure 1: Structure of Skip-gram with $|D| = 10.000$ and $d = 300$
mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model

In its simplest form SGNS can be described as a neural network that given an input word w tries to predict a word c which appears in the context of w. In this case, the context of a word is

defined by a symmetric window of a certain size. If the window size is 1 for example, only the words to the left and the right of the word in a text are considered its context. For our experiments we used a window size of 10. A size this large is very common Schlechtweg et al. (2019); Ferrari et al. (2017); Shoemark et al. (2019) and ensures that for a word in a sentence all other words in the sentence are counted as context, unless the sentence is very long. The internal structure of the neural network is an input and output layer with $|D|$ nodes, see Figure 1.

The in- and outputs of the network are one hot vectors, i.e. the i-th word in the Dictionary D is represented with $|D| - 1$ zeros and a 1 at the i-th node. Between the input and output layer is a single hidden layer with d nodes. The weights connecting the input layer to the hidden layer are stored in the *word matrix*, which is used to get the word vectors. The second sets of weights, which connect the hidden layer and the output layer are stored in the *context matrix*. This matrix is only relevant during the training of the model. Each word in D has a d-dimensional vector in the word matrix. The objective of the SGNS model remains to predict the context of words, but once the model is fully trained on a corpus we are only interested in the weights contained in the word matrix. To reduce computation complexity during the training, a method called Negative Sampling is used. The word and context vectors within their respective matrices solve

$$\arg\max_{\theta} \sum_{(w,c)\in D} \log \sigma(v_c \cdot v_w) + \sum_{(w,c)\in D'} \log \sigma(-v_c \cdot v_w), \tag{1}$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$, D is the set of all observed word-context pairs and D' is the set of randomly generated negative samples (Mikolov et al., 2013a;b; Goldberg and Levy, 2014). The optimised parameters θ are v_{w_i} and v_{c_i} for $i \in 1, ..., d$. D' is obtained by drawing k contexts from the empirical unigram distribution $P(c) = \frac{\#(c)}{|D|}$ for each observation of (w, c), cf. (Levy et al., 2015). After training, each word w is represented by its word vector v_w. To keep our results comparable to previous research (Hamilton et al., 2016b; Schlechtweg et al., 2019) we chose common settings for most of the hyper-parameters. We decided on an initial learning rate α of 0.025, number of negative samples $k = 5$ and no sub-sampling. As we focus on the effect of dimensionality, each experiment was done with d = {5, 10, 25, 50, 80, 150, 200, 250, 300, 350, 500, 750, 1000}.

4.2 Alignment

Alignment methods are needed, because when comparing vectors from different spaces it is important that the columns represent the same coordinate axes. Aligned axes may not be given due to the stochastic nature of dimensional word representations (Hamilton et al., 2016b).

Vector Initialisation (VI) In VI we first train the SGNS model on one corpus and then use the word and context vectors to initialise the vectors for training on the second corpus Kim

et al. (2014).The motivation of this procedure is that if a word is used in similar contexts in both corpora, the second training step will not change the initial word vector much, while differentiating contexts will lead to a greater change of the vector.

The performance of this model is influenced by training order. This is very prominent with corpora of different sizes. It is advisable to first train on the bigger corpus, followed by the smaller. If the training order was switched, i.e. first trained on c_2 and then c_1, it is indicated by the addition of '_BW' after VI.

Orthogonal Procrustes (OP) SGNS is trained on each corpus separately, resulting in matrices A and B. To align them we follow Hamilton et al. (2016b) and calculate an orthogonal orthogonally-constrained rotation matrix W^*:

$$W^* = \arg\min_W \|BW - A\|^2 \tag{2}$$

where the i-th row in matrices A and B correspond to the same word. Using W^* we get the aligned matrices $A^{OP} = A$ and $B^{OP} = BW^*$. Prior to this alignment step we length-normalize and mean-center both matrices (Artetxe et al., 2017; Schlechtweg et al., 2019).

Word Injection (WI) The sentences of both corpora are shuffled into one joint corpus, but all occurrences of target words are substituted by the target word concatenated with a tag indicating the corpus it originated from (Ferrari et al., 2017; Schlechtweg et al., 2019; Dubossarsky et al., 2019). This leads to the creation of two vectors for each target word in one vector space, while non-target words receive only one vector encoding information from both corpora.

4.3 Change Measure

Cosine Distance (CD) was used to quantify differences between vectors. It is based on Cosine Similarity (Salton and McGill, 1983).

$$CD(\vec{x}, \vec{y}) = 1 - \frac{\vec{x} * \vec{y}}{||x||_2 * ||y||_2} \tag{3}$$

The latter part of the subtraction in Equation 3 is the cosine of the angle between the two non-zero vectors $\vec{x}$ and $\vec{y}$. Subtracting it from one makes identical vectors have a CD of 0, and unrelated vectors a CD of close to 1.

5 Experimental Setup

5.1 Data

Our data was provided by the SemEval-2020 Task 1 (Schlechtweg et al., 2020). The organisers of the task hosted a competition at `https://competitions.codalab.org/competitions/20948`. Participants were able to compare their models on two subtasks of SCD. The first subtask is a binary classification task with the second being a ranking task, which is the one we use to evaluate the three models. For each of the four languages (English, German, Latin and Swedish) a corpus c_1, a corpus c_2 and a ranked list of target words is provided.

5.1.1 Corpora

Table 1 shows basic corpus statistics. All corpora are lemmatised and without punctuation. The SemEval corpora are samples from CCOHA Davies (2012); Alatrash et al. (2020), DTA Deutsches Textarchiv (2017), BZ Berliner Zeitung (2018), ND Neues Deutschland (2018), LatinISE McGillivray and Kilgarriff (2013) and KubHist Språkbanken (Downloaded in 2019). *Tokens* states the number of total words within the corpora, *types* states the number of individual words ($|D|$). Differences between corpora are not only limited to language, but extend to different time periods, corpus size and Type-Token ratio (TTR).

	C_1	C_2	tokens_1	tokens_2	types_1	types_2	TTR_1	TTR_2
English	CCOHA 1810–1860	CCOHA 1960–2010	6.5M	6.7M	87k	150k	13.38	22.38
German	DTA 1800–1899	BZ+ND 1946–1990	70.2M	72.3M	1.0M	2.3M	14.25	31.81
Latin	LatinISE -200–0	LatinISE 0–2000	1.7M	9.4M	65k	253k	38.24	26.91
Swedish	Kubhist 1790–1830	Kubhist 1895–1903	71.0M	110.0M	3.4M	1.9M	47.88	17.27

Table 1: Corpus statistics. TTR = Type-Token ratio (number of types / number of tokes * 1000).

5.1.2 Gold Data

Creating change scores based on human annotations is very time consuming and labour intensive and requires specialised methods for large data sets. The following paragraph describes the approach taken by Schlechtweg et al. (2020). For each target word, use pairs are extracted from c_1 and c_2. Use pairs are two sentences containing the same target word. Human judges have to score on a scale from 0 to 4, how similar the meaning of the target word in both sentences is. From this a graph is constructed, with the use pairs as nodes and the scores as the weight connecting the use pairs. The nodes of the graph are grouped into clusters. Nodes within the

same cluster have appeared in similar meanings. To avoid needing human judges to score all $n * (n-1)/2$ edges between n nodes, only the most important edges are weighted with scores. For example, if the pairs (p_1, p_2) and (p_2, p_3) have a high similarity score, the pair (p_1, p_3) should also have a high similarity score. The graph and its clusters are updated over several iterations and in each iteration only a few pairs are annotated by humans. The end product is a graph for each target word, in which each cluster represents a sense. From this a sense frequency distribution are created, see Table 2 as a reference for the words *cell* and *tree*. See Schlechtweg et al. (2020) for a detailed description of the annotating process. The Jensen-Shanon distance is used as a metric of semantic change between the sense frequency distribution of t_1 and t_2.

The set of target words contains words of varying change scores. Some words are very stable and have no semantic change between t_1 and t_2, while others change significantly. Table 2 provides an example for the sense frequency distribution for the words *cell* and *tree*. Both words have acquired a new sense in t_2. Yet the degree of semantic change of these words is very different. The senses of the word *cell* shift from *Chamber* and *Biology* to mainly *Phone* and *Biology*. *Chamber* is almost completely lost as a sense in t_2. However for the word *tree*, *Botany* remains a very dominant sense. The newly acquired sense in *Computing* is limited to very technical literature and thus only makes up a small part in the sense frequency distribution.

The difficulty of this task is that the rankings created by the models have to be the same as the rankings in the gold data. How well the models performed at this task is calculated using the Spearman's rank correlation coefficient. A high coefficient indicates a high correlation between the model predictions and the gold ranking. For the previous example the models should rank the semantic change of *cell* higher than the one of *tree*.

	cell			*tree*	
Senses	Chamber	Biology	Phone	Botany	Computing
# uses in t_1	12	18	0	30	0
# uses in t_2	1	11	18	29	1

Table 2: Sense frequency distribution of *cell* and *tree* in t_1 and t_2 (Schlechtweg et al., 2020). The depicted data is for demonstration purposes only. True sense frequency distribution of these words may differ.

5.2 Experiment 1: Optimal Dimensionality

Our first experiment is aimed to answer our four initial hypotheses. Figure 2 shows the performance of all models on the four languages across dimensionalities. Each evaluation run was performed five times with identical parameter settings. This way we are able to detect variance

and take the mean as a representative value. In Figure 2 this is visualised by the bars showing the minimal and maximum value. The continuous line shows the mean. Depending on corpus size we trained the model for either 5 (German, Swedish) or 30 epochs e (English, Latin).[2] The number of training epochs determines how often the model iterates over the corpus. This can be used to artificially increase training data.

For the analysis of the first three hypotheses we will focus on the results of German and Swedish performances as they are better overall and have less variance (indicated by the length of the bars in Figure 2). The Swedish and Latin corpora had significant size differences where c_1 was smaller than c_2. We have observed that the performance of VI can be improved by first training on the larger corpus and afterwards on the smaller corpus. We attribute this behaviour to the quality of the weights used for initialising the second training process. A larger corpus generally leads to better semantic representation of words and thus the quality of the weights used for initialisation is better.

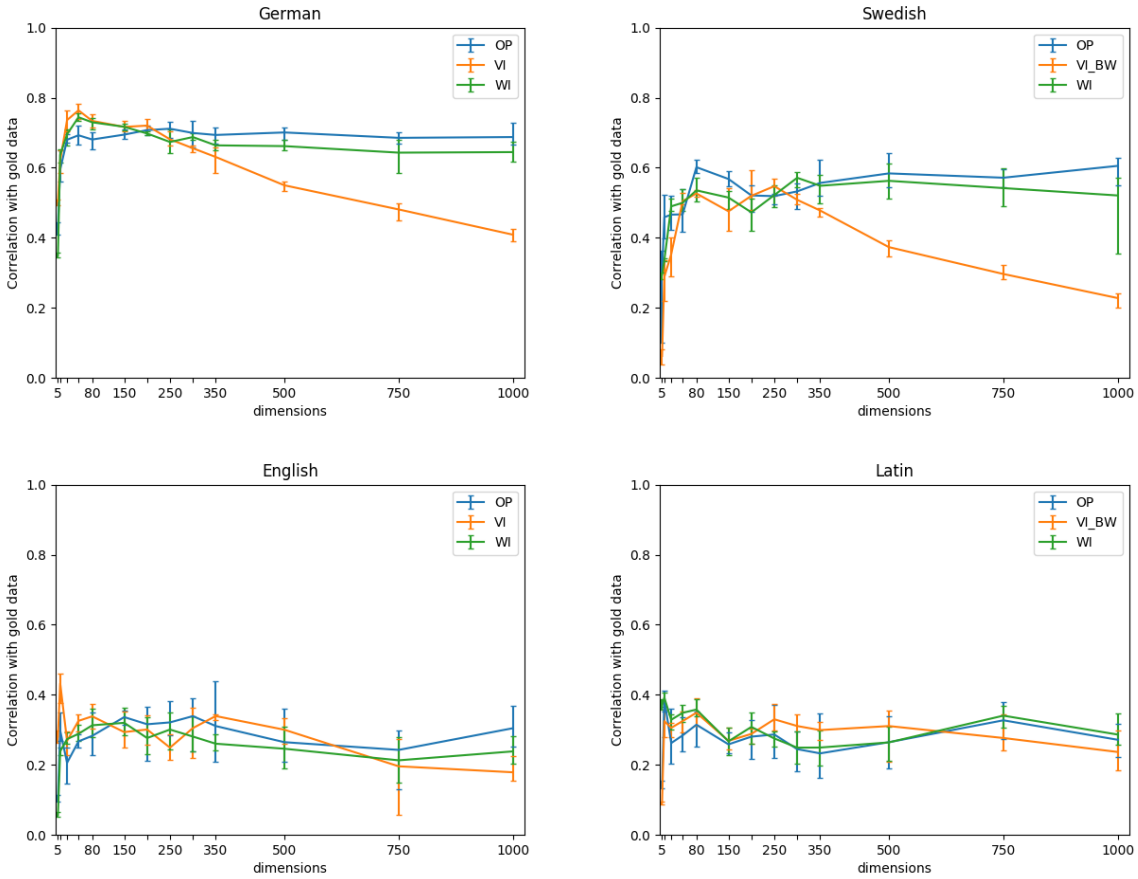


Figure 2: Comparing all models with varying dimensionality. The line refers to the mean of 5 runs; error bars show the max/min values.'_BW' indicates switched training order.

[2]We tried alternative numbers of epochs with mixed results.

(1) The optimal dimensionality is different for all three models. In German VI and WI have their optimal d at 50 and OP has its at 250. Yet, for OP there is no distinct peak and becomes very stable once d is greater or equal to 50. For Swedish, OP has its peak at d=100 according to the averaged results (best performance out of all runs is at d=500, but with high variance). VI and WI also have a local maximum at d=100 but the global maxima are reached between 200 and 300. Therefore, making a decisive statement about different optimal dimensionalities is not possible. However it can be observed that the optimal d for VI and WI often are in each other's proximity.

Table 3 lists the best performances of the three models on all four languages. These scores are not averaged across multiple runs but rather the overall maximum scores. Entries are annotated with the d/e used to achieve the score. For English and German, VI and WI had the same optimal d. For Latin and Swedish, OP and WI had the same optimal d. Overall peak performances of the models also seem to sometimes be at identical d.

Model	AVG	English	German	Latin	Swedish
VI	**.58**	**.46** *10 / 30*	**.78** *50 / 5*	.39 *80 / 30*	**.67** *300 / 10*
OP	.56	.44 *350 / 30*	.73 *300 / 5*	**.41** *10 / 30*	.64 *500 / 5*
WI	.54	.36 *10 / 30*	.76 *50 / 5*	.41 *10 / 30*	.61 *500 / 5*

Table 3: Performances after optimising d and e. Best scores for each category are bold and annotated with used (optimal) d / e

(2) VI has a lower optimal dimensionality than OP, and OP has a lower one than WI. The plots disprove this statement. As previously described, VI and WI have very similar optimal dimensionality. For German, the optimal d of OP is higher than the one of VI/WI and in Swedish it is lower. For VI and WI d only influences the sensitivity of the model. Increasing d past the optimum makes these models too sensitive and thus more susceptible to noise. OP on the other hand may benefit from very high d, due the alignment method having more degrees of freedom.

(3) VI captures more noise than OP and OP captures more noise than WI with equal dimensionality. For this experiment we need to measure noise. We regard any change the models report that does not originate from true semantic change across the two corpora as noise. To differentiate between signal (true semantic change) and noise we followed Dubossarsky et al. (2017) and created corpora where target words do not have any semantic change between them.

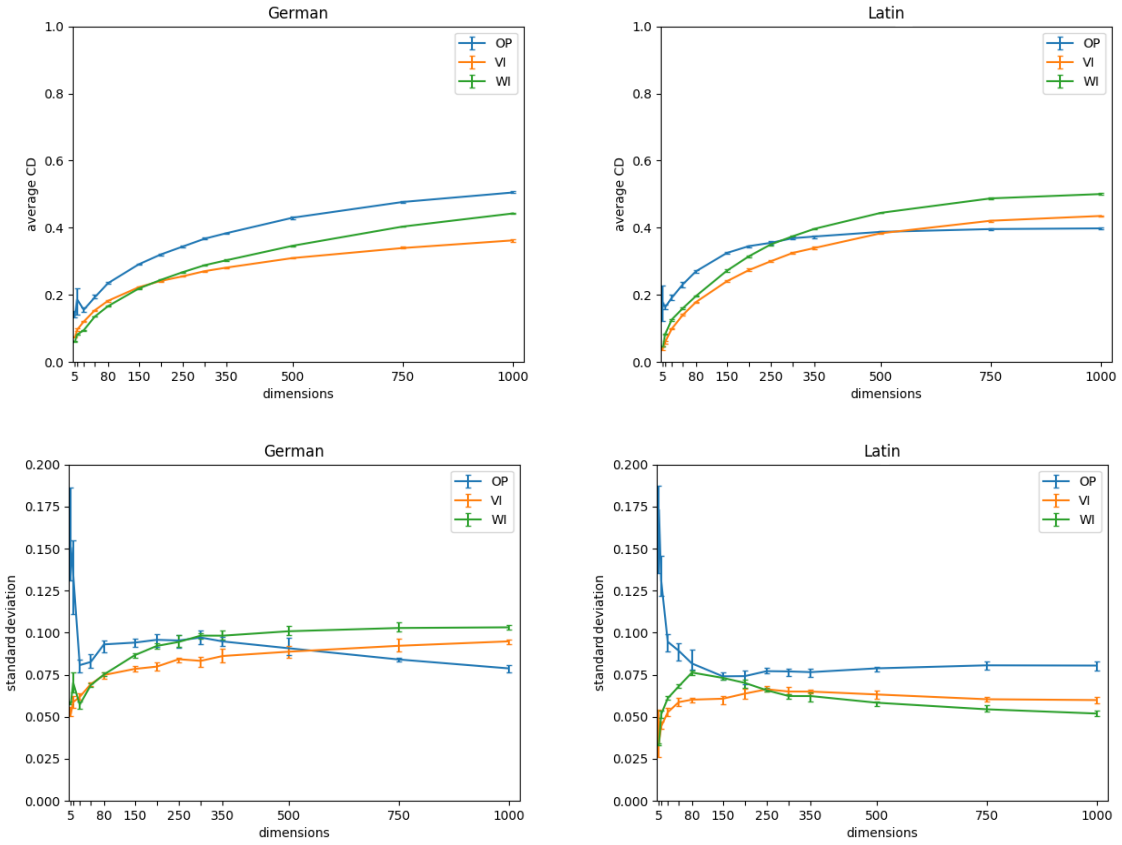


Figure 3: **Top** figures show the noise level on German and Latin shuffled corpora. **Bottom** figures show variance within the noise.

This is done by first mixing all sentences from c_1 and c_2 into one big corpus in random order. In the second step the big corpus is split into two corpora, $\tilde{C_1}$ and $\tilde{C_2}$. Corpus size and target word frequencies can influence the quality of word embeddings. If they were to change during shuffling it might introduce a new variable. That is why we preserved them during the shuffling process, i.e., C_1 and $\tilde{C_1}$ contain the same number of sentences and all target words have the same number of occurrences in C_1 as they have in $\tilde{C_1}$, the same is true for C_2 and $\tilde{C_2}$.

As mentioned above, to measure noise we used shuffled corpora which should contain close to no semantic change between them. Any change the models report on these corpora can be regarded as noise. The gold score for each word is 0. It is pointless to calculate the spearman correlation of the ranking and therefore we calculated the average CD and the standard deviation. The average CD can be interpreted as the level of noise. For these measures the plots are given for German and Latin as they are very similar to Swedish and English respectively. Looking at Figure 3 top, an overall increase in the level of noise can be observed with increasing dimensionality. This is coherent with Yin and Shen (2018). Concerning the differences between models, for the most part the curve of VI is lower than the curve of WI. OP is depending on language and d, lower or higher than the other two models.

It is important to note that performance may still be heavily impacted by noise even if the noise level is quite low, see German VI with high d. Or Vice versa, high noise level and good performance, see German OP with high d. This can be explained by the noise level not influencing the rankings as much as say variance in the noise. A high average can come without much variation. We used the standard deviation as a second measure to isolate the variations of the noise. The rankings may change if some words have a higher or lower amount of noise added to them. The bottom of Figure 3 shows the plots for the results we got from this measurement. OP has the interesting tendency of a decrease in variation of noise with increasing d. One explanation for this behaviour could be that the alignment works better with higher d as it allows for more degrees of freedom for the rotation Matrix. Again VI and WI behave similarly, however on German WI has a slightly higher standard deviation, as well as for the lower d in Latin. For d greater than 250 VI has a higher deviation.

This data leads to the conclusion that this hypothesis is wrong as well. We hoped to find a stronger connection between noise and optimal d, which would allow us to predict the optimal d without tuning on gold data. This lack of connection is very prominent with VI. In 2 we see a strong drop in performance, compared to OP and WI, with higher d, in all languages. This is not transferred as a significant increase in level or variance of noise of VI compared to the other models. The main reason for a drop in performance with high d is noise. Higher d allows for more information within the embeddings. There is a certain point where the embeddings contain the highest amount of semantic information without picking up more noise than necessary. Increasing d past this point leads to changes measured by VI representing less actual semantic change and more noise. It seems that the average and standard deviation of noise is not capable of quantifying this kind of noise. In the following experiments we will focus on word frequency as a source of noise.

(4) The optimal dimensionality for each model is a function of other parameters such as number of training epochs and corpus size. The explanation of the connection between number of training epochs and optimal dimensionality will be subject of the next two experiments as there is an interesting and unexpected interaction of frequency noise and the number of training epochs.

The differences between the corpora of the SemEVAL-2020 Task 1 are not just limited to language. According to the number of tokens for all languages in Table 1 there are very significant size differences between corpora. The optimal dimensionality for each language could give an answer to this hypothesis. However the performance of all models on English and Latin is very inconsistent and lower than the performance on German or Swedish with no clear optimum. Apart from size, the corpora also differ in homogeneity and type-token ratio, making the isolation of corpus size more difficult. An approach to solve this problem could have

been to work with one language and incrementally decrease the size of C_1 and C_2, by removing some sentences. Under consideration of the target word frequency this method should provide an answer to the hypothesis. Due to time constraints we were not able to conduct experiments with this methodology. Relying only on the results we have, the point could be made that corpus size influences optimal d. In English and Latin we observe performances at d=10 which are already very close to the maximum. For German and Swedish that point is reached around d=80. Additionally in Swedish we often see two maxima. These could be caused due to the two corpora in Swedish having different optimal d, c_1 is approximately half the size of c_2. These results should only be taken as inspiration to further investigate this hypothesis.

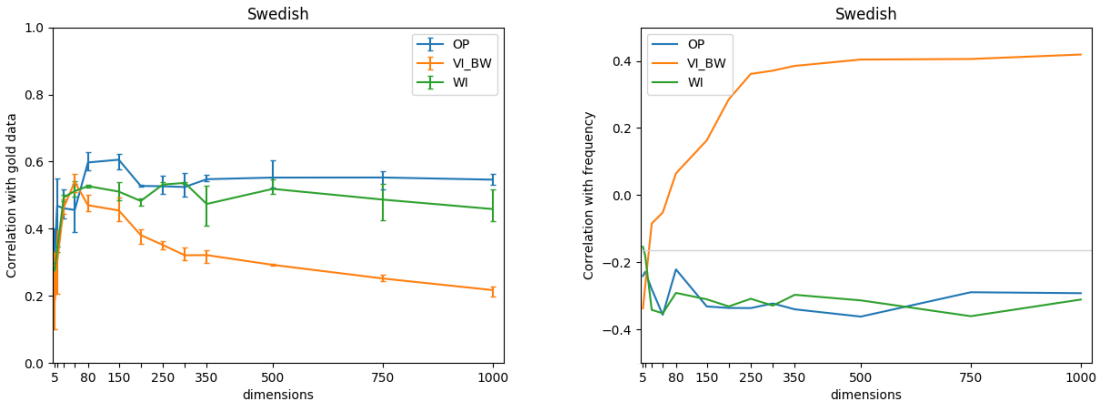
Additional Notes; Looking at Figure 2 it is noticeable that the optimal dimensionality is often lower than the common choice of d=300 (Schlechtweg et al., 2019; Hamilton et al., 2016b; Mikolov et al., 2013a), even for OP and WI. Indicating that in most research of LSCD d could have been chosen much lower to reduce model complexity and even an increase in performance.

The data we used was part of the SemEval-2020 Task 1 so we have access to the performance reached by other state-of-the-art models. Compared to these, our best results of the three models are within the Top-5 during the post-evaluation phase. The best performance of VI is second in total and first for German and Swedish. With tuning on d, and e in the case of VI, these models are amongst the best for SCD. [3]

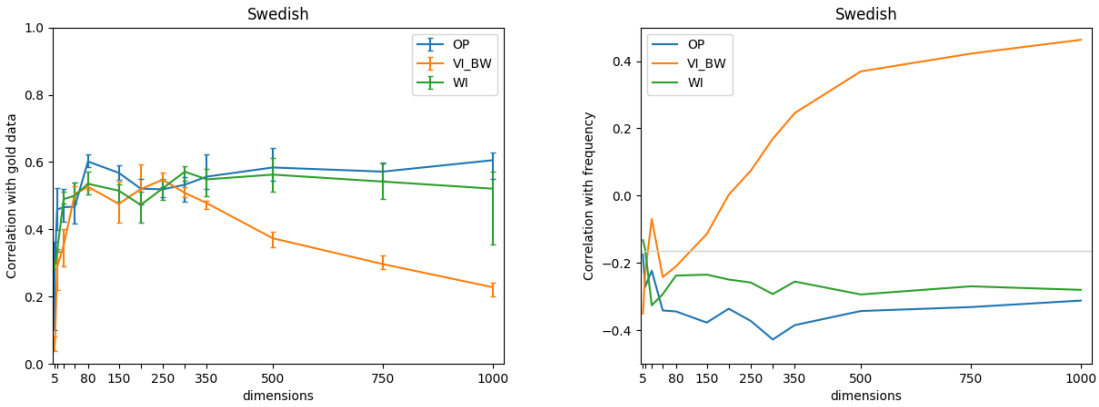
5.3 Experiment 2: Frequency Bias

We have observed very good performance of VI with low d, indicating that the model's complexity is sufficient to capture semantic differences over time. Thus, the only reason for a performance drop with higher dimensionalities is that non-semantic properties are picked up by the model. We regard these properties as noise. As indicated by observations in Schlechtweg et al. (2019) VI is sensitive to the number of updates done in the second training step. This conforms with the previous observation that training order can influence performance. We computed the correlation between the target word ranking of CD and word frequency in the second corpora. The influence of updates to the vectors in the second training step should be reflected in this correlation. We empirically verified that the frequency component that matters is the frequency in the second corpus. Other frequency components like, frequency in the first corpus or frequency differences between corpora have no strong influence. From now on we just use *frequency* to refer to the frequency in the second corpus. Those cases in which the training order was switched, the second corpus is C_1.

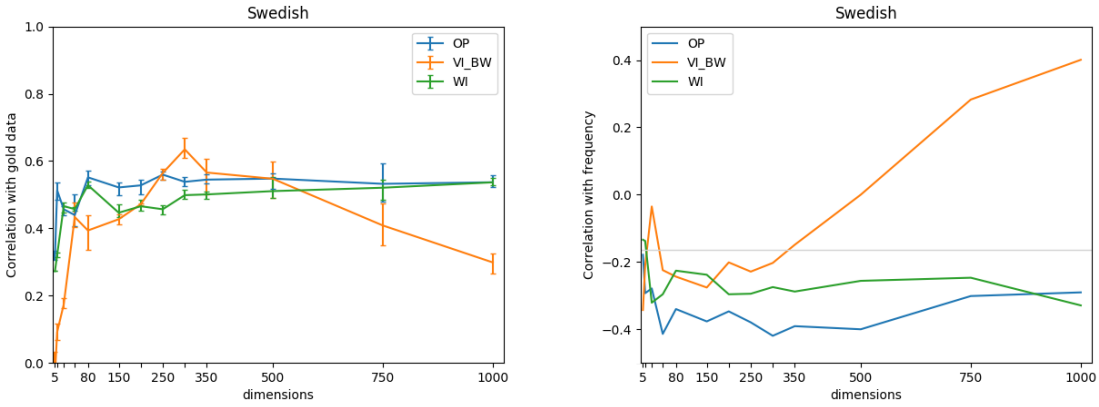
[3] `https://competitions.codalab.org/competitions/20948` Team *in vain*



2 epochs



5 epochs



10 epochs

Figure 4: **Left** figures show performance on Swedish data for Subtask 2 across d, with increasing numbers of epochs. **Right** figures show correlations between CD and frequency in Swedish corpus_2. Gray lines indicate true correlation in gold data.

Figure 4 right depicts the frequency correlation across different d. The Gray line indicates the true correlation in the gold data. OP and WI show no anomaly regarding correlation with frequency and have steady values. VI on the other hand shows a very distinct increase in correlation with d. The figure has plots for different numbers of training epochs (2, 5, 10). Comparing the frequency correlations (right) and performance (left) it is clearly visible that the two are negatively correlated. Once the frequency correlation encoded in the CD rankings surpasses a certain point, the performance begins to drop. Although only the plots for Swedish are shown, this frequency bias is observable for the other languages too, suggesting that it is an inherent effect in VI. There should be no correlation between frequency and semantic change greater than the true correlation within the gold data. The correlation that is present in the gold data is deliberate as the semantic change of words is unrelated to their frequency Dubossarsky et al. (2017). Therefore we can confirm that this frequency bias of VI can be considered as noise. With this insight and deliberate consideration as a measure of noise, the hypothesis about VI picking up the most amount of noise among the other models is true. Neither average nor standard deviation on the shuffled corpora could measure frequency noise to this degree. This would suggest that there is a need to specify the noise we are interested in when making hypotheses like we have. The differences between the frequency correlation and the other two measures of noise is that with frequency correlation the noise level is unknown. If the frequency noise level is relatively low it could explain why it was not visible with the average and standard deviation.

(4) the optimal dimensionality for each model is a function of other parameters such as number of training epochs and corpus size. cont. Figure 4 bottom shows the performance on Swedish with different e. For OP and WI we see no significant changes in optimal dimensionality, apart from slight deviations. Optimal d for VI on the other hand, is controlled by e. This is due to the frequency bias being more or less prominent in lower d in regards to the choice of e. The main source of noise for VI in high d is the frequency bias, hence we can directly control the optimal d with e.

5.4 Experiment 3: Reducing the Frequency Bias

Increasing e: As already indicated in Figure 4 the frequency bias is influenced by the number of training epochs e. Where an increase in e reduces the frequency correlation with low d. Training with more epochs led to VI reaching a higher performance in Swedish than OP and WI, even at d=300. This approach has the negative side effect that the time needed to train the SGNS model linearly increases with e. Furthermore, while it reliably dampens the frequency bias, it does not always make the model perform better. For example with German the optimal d went from 50 to 250 by increasing e from 5 to 10. However it was found that the higher optimal d of

250 did not perform better than the one at 50 (see Figure 5 left, blue and red line). To further analyse this behaviour, we split the parameter number of training epochs e into two separate parameters e_1 and e_2. When training the SGNS model with VI alignment we have two training steps. Until now we used the same number of training epochs for both training steps. For the following experiments we used e_1 training epochs for the first step and e_2 training epochs for the second training step. With this method we aimed to isolate the parameter that actually influences the behaviour we observed.

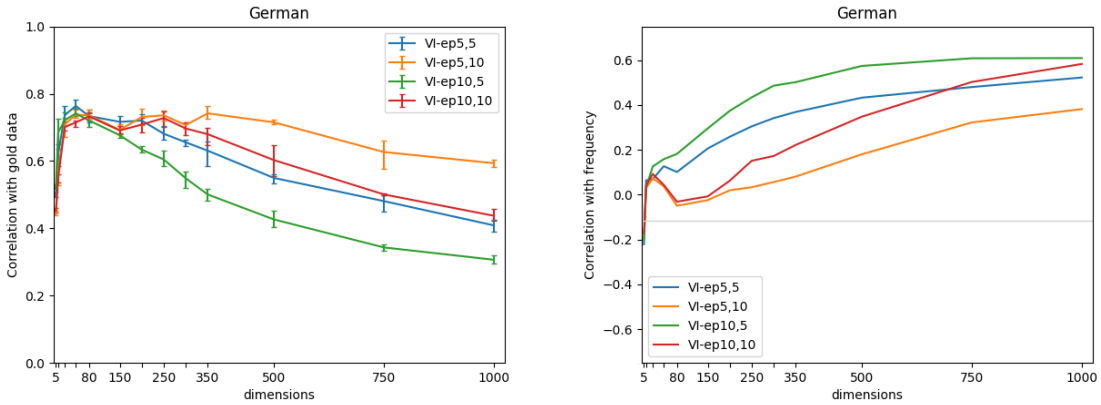


Figure 5: Comparing performance (**left**) and frequency correlation (**right**) between different epoch settings. "ep x, y" indicates that x epochs were used during the first training step, and y epochs during the second training step.

We tested four combinations with $e_1, e_2 \in \{5, 10\}$, for $e_1 = e_2 = 5$ the parameters are identical to the ones used in Section 5.2. Figure 5 displays performance and frequency correlation for German of VI. OP and WI are not shown as we neither observed a frequency bias, nor significant changes with different e for them. The performance of all parameter settings look very similar up until $d = 150$, from there on the differences become more clear. The combination of $e_1 = 5, e_2 = 10$ has the best performance with higher d. This is also reflected in the correlation with frequency (right) as it has the lowest across all dimensions. From these plots we can make the observation that the frequency bias depends on the initialised vectors and the amount of updates done to the vectors during the second training step. The initial hypothesis about the frequency bias, prior to experimenting with different e, was that the fewer updates to the vectors the better. The weights used for initialisation should already capture semantic relations between words very well as they have previously been created by training on one of the corpora. Note that with alignment methods like OP, the SGNS model trains on the corpora separately, so the quality of the created embeddings are equal to the ones used for initialisation of VI. Any updates to the weights during the second training step are thus "unnecessary", unless they stem from semantic change. Our intuition was that by reducing the number of training epochs the number of "unnecessary" updates gets reduced and should help to dampen the frequency bias. However

this hypothesis does not fit to the results we observe here. The opposite is true, more training epochs help to make the updates to the vectors reflect semantic change more than they reflect frequency.

After analysing the results, new hypotheses that could explain the phenomenon were formed. (1) During the training phase of SGNS, word vectors eventually reach a state, where they represent the word as closely as possible. We will refer to this state as the "final state". Once the final state is reached, updates to the vector are very small and the values of the vector remain almost constant. (2) Depending on the dimensionality of the model and frequency of the word, the final state is reached at different time steps in training. (3) Higher dimensionality and higher word frequency lead to the final state being reached earlier.

This hypothesis fits the observed data so far. The idea is that by increasing e, more updates are done to the vectors. These updates will not affect low-dimensional and vectors of high frequency words much, as they have already reached their final state. High-dimensional vectors and vectors of low frequency words on the other hand will benefit from more updates, as they are able to reach their final state. If e is large enough, all vectors should have reached their final state and thus represent the words as accurately as possible. At the time of the establishment of this hypothesis, the scope of this thesis was already quite large and we did not have enough time to systematically find approaches to confirm or disprove the hypothesis. The following experiment shows very interesting results but their repercussion regarding the cause of the frequency bias remains unexplored.

Frequency groups: For this experiment we split the target words into three frequency groups and looked at performance and frequency bias of these groups. The thresholds for the three frequency groups are 200 and 1000. Low frequency contained 24 words, mid frequency 14 words and high frequency 10 words. We chose $e_1 = 5, e_2 = 5$ and $e_1 = 5, e_2 = 10$ to observe how differences in training epochs for the second training step affect performance. The results are shown in Figure 6. For this experiment only the German data set was examined. Statements on the results are specific to German and might differ for other data sets.

Against intuition the low frequency group has the best performance followed by mid and then high. We expected low frequency words to perform the worst as their frequency in C_1 was also quite low, thus the training samples for the words are smaller. The frequency bias is mostly contained in the mid frequency group, but a slight increase in frequency correlation can also be observed in the low frequency group. The true correlation in the gold data for the low, mid and high frequency groups are -0.22, -024 and -0.02. Switching from $e_2 = 5$ to $e_2 = 10$ mostly impacts performance of the mid and high frequency group at lower d. The low frequency group seems not to have changed. Comparing the frequency bias, we see that the biggest reduction is in

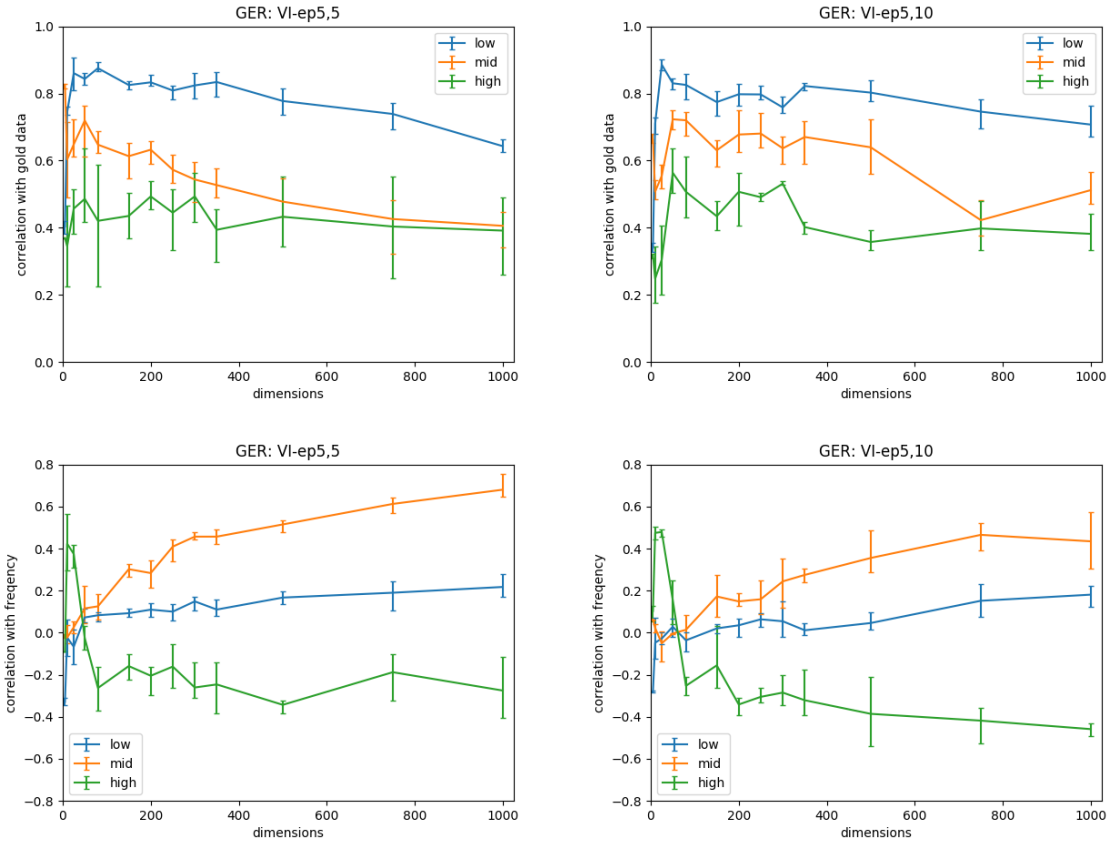


Figure 6: **Top** figures compare the correlation within different frequency groups for $e_2 = 5$ and $e_2 = 10$. **Bottom** figures show the frequency correlation.

the mid frequency group. From this we can confirm that an increase in training epochs for the second training step improves performance of some frequency groups, while other frequency groups remain unaffected. The data for high-frequency words also shows that despite the absence of frequency correlation, low performance is possible.

Length normalisation: This last experiment also shows an interesting interaction with the frequency bias. Aside from the explanation used to motivate the experiment we could not further explain these results. This experiment was motivated by Schakel and Wilson (2015), where a relation between word frequency and vector length is found. Their findings are that low and high frequency words tend to have shorter vectors. Vector length is measured using the L2 Norm. The intuition was that vector length influences the magnitude of the updates done to the vectors during the second training step. For this experiment we length-normalise the word vectors between the two training steps. Results are visualised in Figure 7. We compared performance and frequency correlation of genuine VI and the version using L2 normalisation of the word vectors used for initialisation. Note that for VI both *word vectors* and *context vectors* are used to initialise the second model. Length normalising the context vectors yields no befit regarding

performance or frequency correlation. Length normalising the word vectors seems to be the most promising method to fight the frequency bias and get promising results even when using high d. After VI genuine reached its peak performance it drops off rapidly, unlike VI with length normalisation where performance stays very stable. This improvement at higher d does not come with a negative influence on performance at lower d. A slight increase in frequency correlation with d is still visible but this correlation does not exceed 0.2.

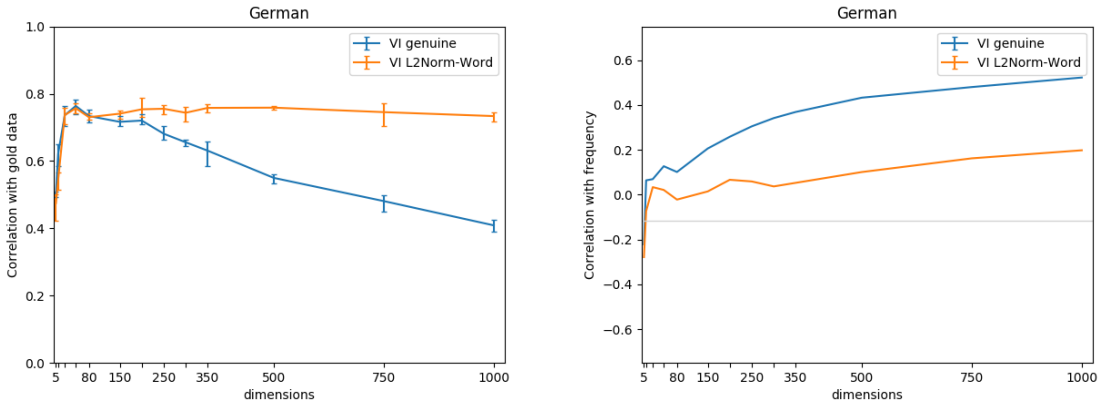


Figure 7: Comparing performance (left) and frequency correlation (right) between genuine VI and VI with l2 normalisation of word vectors used for initialisation.

6 Discussion and Conclusion

The work detailed in this thesis analysed the importance of dimensionality in LSCD in relation to noise. Three popular models with expected different susceptibility to noise were compared to each other. Answering the initial hypotheses was more difficult than expected. The first problem was the assumption of an optimal dimensionality. Often the optimum was not eminently clear. For example OP often had very consistent performance on the German data set, regardless of d(if high enough). English and Latin had substantial variances in performance with equal parameters. This drastically reduced the data we could use to address the hypotheses. Additionally for VI optimal d is dependent on e. While optimal d of OP and WI is mostly independent of e. Thus claims made on the optimal d for the three models are dependent on e (Hypothesis 1 and 2).

The results of the conducted experiments yield new insights regarding the choice of dimensionality. Previous work followed the recommendation by (Mikolov et al., 2013a;b; e.g.) and used a dimensionality of 300. Although we could not link optimal dimensionality to corpus size due to inconsistent and sub-optimal performance with smaller corpora (English and Latin) we still think the two could be linked. Corpora used in LSCD are very small compared to other corpora used for tasks like word similarity measures. If corpus size and optimal dimensionality

are linked, lower d could significantly reduce computational complexity and in some cases even improve overall performance.

The results reported in Section 5.2 often stated an optimal d lower than 100 for all three models. For OP and WI a higher choice did not impact performance negatively, but VI is very sensitive regarding d. The overall sensitivity of VI regarding parameter choice could explain why in previous research performances for VI were inferior to other state-of-the-art models. Yet, once VI is properly tuned regarding parameters like; (1) training order, (2) dimensionality d and (3) number of training epochs e it is capable of achieving higher correlation scores than OP or WI. Finding the best parameters in a setting without knowledge of true semantic change remains very challenging. The goal is to maximise signal while minimising noise, i.e. reaching the highest signal-to-noise ratio. The approach of shuffling the corpora to measure noise, either average change scores or the standard deviation between change scores, in order to find the highest signal-to-noise ratio was unsuccessful. This method of measuring noise needs more improvements. For VI we needed a different approach to link the performance drop to noise. It is possible other specialised methods need to be used in order to maximise the signal-to-noise ratio without gold data. Even under consideration of the correlation with frequency as noise, the main observation was that noise increases with dimensionality. Which is not surprising but it is also important to know how complex the model needs to be (dimensionality) in order to encode important semantic properties. The higher the dimensionality of the vectors, the finer details can be picked up. So the optimal dimensionality which maximises the signal-to-noise ratio is in between the two extremes. The solution of this problem is either to have some method of finding the parameters that maximise the signal-to-noise ratio according to the model and data set, or to have very robust models. OP and WI are examples for robust models, their performance is only marginally influenced by dimensionality (if chosen high enough), number of training epochs and their approach for alignment has no explicit training order which may need to be changed.

With the insights gained from Experiment 3 we seem to have found methods which make VI more robust. This includes increasing e and the length normalisation of the word vectors used to initialise the second model. Though it is still unclear why these methods reduce the influence of the frequency bias. Knowing that VI is sensible to frequency differences it is advisable to decide on training order based on corpus size and or differences in target word frequency distribution.

References

Adnan Ahmad, Kiflom Desta, Fabian Lang, and Dominik Schlechtweg. Shared task: Lexical semantic change detection in German (Student Project Report). *arXiv:2001.07786*, 2020.

Reem Alatrash, Dominik Schlechtweg, Jonas Kuhn, and Sabine Schulte im Walde. CCOHA: Clean Corpus of Historical American English. In *Proceedings of the 12th International Conference on Language Resources and Evaluation*, Marseille, France, 2020. European Language Resources Association.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 451–462. Association for Computational Linguistics, 2017.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1023. URL https://www.aclweb.org/anthology/P14-1023.

Berliner Zeitung. Diachronic newspaper corpus published by Staatsbibliothek zu Berlin, 2018. URL http://zefys.staatsbibliothek-berlin.de/index.php?id=155.

Mark Davies. Expanding Horizons in Historical Linguistics with the 400-Million Word Corpus of Historical American English. *Corpora*, 7(2):121–157, 2012.

Deutsches Textarchiv. Grundlage für ein Referenzkorpus der neuhochdeutschen Sprache. Herausgegeben von der Berlin-Brandenburgischen Akademie der Wissenschaften, 2017. URL http://www.deutschestextarchiv.de/.

Haim Dubossarsky, Daphna Weinshall, and Eitan Grossman. Outta control: Laws of semantic change and inherent biases in word representation models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1147–1156, Copenhagen, Denmark, 2017.

Haim Dubossarsky, Simon Hengchen, Nina Tahmasebi, and Dominik Schlechtweg. Time-Out: Temporal referencing for robust modeling of lexical semantic change. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 457–470, Florence, Italy, 2019. Association for Computational Linguistics.

Alessio Ferrari, Beatrice Donati, and Stefania Gnesi. Detecting domain-specific ambiguities: An NLP approach based on wikipedia crawling and word embeddings. In *Proceedings of the 2017 IEEE 25th International Requirements Engineering Conference Workshops*, pages 393–399, 2017.

Yoav Goldberg and Omer Levy. Word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv:1402.3722*, 2014.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. Cultural shift or linguistic drift? Comparing two computational measures of semantic change. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2116–2121, Austin, Texas, 2016a. Association for Computational Linguistics.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1489–1501, Berlin, Germany, 2016b. Association for Computational Linguistics.

Zellig S. Harris. Distributional structure. *WORD*, 10(2-3):146–162, 1954.

Nobuhiro Kaji and Hayato Kobayashi. Incremental skip-gram model with negative sampling. *CoRR*, abs/1704.03956, 2017. URL `http://arxiv.org/abs/1704.03956`.

Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. Temporal analysis of language through neural language models. In *LTCSS@ACL*, pages 61–65. Association for Computational Linguistics, 2014.

Omer Levy, Yoav Goldberg, and Ido Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225, 2015.

Barbara McGillivray and Adam Kilgarriff. Tools for historical corpus research, and a corpus of Latin. In Paul Bennett, Martin Durrell, Silke Scheible, and Richard J. Whitt, editors, *New Methods in Historical Corpus Linguistics*, Tübingen, 2013. Narr.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119, Lake Tahoe, Nevada, USA, 2013b.

Neues Deutschland. Diachronic newspaper corpus published by Staatsbibliothek zu Berlin, 2018. URL http://zefys.staatsbibliothek-berlin.de/index.php?id=156.

Hao Peng, Jianxin Li, Yangqiu Song, and Yaopeng Liu. Incrementally learning the hierarchicalsoftmax function for neural language models. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3267–3273. AAAI Press, 2017.

Gerard Salton and Michael J McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, New York, 1983.

Adriaan M. J. Schakel and Benjamin J. Wilson. Measuring word significance using distributed representations of words. *CoRR*, abs/1508.02297, 2015. URL http://arxiv.org/abs/1508.02297.

Dominik Schlechtweg, Anna Hätty, Marco del Tredici, and Sabine Schulte im Walde. A Wind of Change: Detecting and evaluating lexical semantic change across times and domains. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 732–746, Florence, Italy, 2019. Association for Computational Linguistics.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain, 2020. Association for Computational Linguistics.

Philippa Shoemark, Farhana Ferdousi Liza, Dong Nguyen, Scott Hale, and Barbara McGillivray. Room to Glo: A systematic comparison of semantic change detection approaches with word embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 66–76, Hong Kong, China, 2019. Association for Computational Linguistics.

Språkbanken. *The Kubhist Corpus*. Department of Swedish, University of Gothenburg, Downloaded in 2019. URL https://spraakbanken.gu.se/korp/?mode=kubhist.

Zi Yin and Yuanyuan Shen. On the dimensionality of word embedding. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 887–898. Curran Associates, Inc., 2018.

A German Summary

In dieser Thesis wird das Thema ”Dimensionalität und Störungen in Modellen der Bedeutungswandel Erkennung” behandelt.

Bedeutungswandel Erkennung: Auf Grund von sozialen und technologischen Einflüssen ändern Wörter ihre Bedeutung im Laufe der Zeit. Diesen Wandel kann man mit speziellen automatisierten Modellen erkennen und quantifizieren. Automatisierte Modelle zur Bedeutungswandel Erkennung bestehen in der Regel aus drei Teilen. 1) Word Embeddings auf Korpus t1 und Korpus t2 erzeugen, 2) Vektorräume der Embeddings angleichen 3) Änderungen zwischen Vektoren messen. Die Modelle müssen für ein bestimmtes Set an sogenannten *Ziel Wörtern* eine Rangliste erzeugen, welche die Wörter nach gemessenem Bedeutungswandel auflistet.

Word Embeddings sind niedrig (ca. 2 bis 1000) dimensionale Vektor Repräsentationen von Wörtern. Wort Vektoren können auf unterschiedliche Weisen erzeugt werden und beinhalten Informationen bezüglich semantischer Beziehungen zwischen Wörtern. In dieser Arbeit verwenden wir Skip-Gram Negative Sampling (SGNS) zum Erzeugen der Embeddings. Dies ist ein viel genutztes Modell in der Maschinellen Sprachverarbeitung, besonders im Feld der Bedeutungswandel Erkennung. SGNS basiert auf einem Neuronalen Netzwerk, welches auf Eingabe eines Wortes versucht dessen Kontext vorauszusagen. Das Training erfolgt auf einem Textkorpus. Wichtige Hyper-Parameter von SGNS, die von uns untersucht werden sind Dimensionalität und Training Epochen Anzahl. Die Dimensionalität bestimmt wie der Name schon sagt, die Dimensionalität der Wort Vektoren. Mit der Training Epochen Anzahl kann man bestimmen wie oft SGNS über den Korpus iteriert. Mehrfache Trainings Durchläufe werden genutzt um die Trainingsdaten künstlich zu vergrößern. Datensätze zur Bedeutungswandel Erkennung bestehen aus zwei oder mehreren Korpora aus verschieden Zeitperioden. Da Word Embeddings auf allen Korpora erzeugt werden, muss darauf geachtet werden, dass die Vektorräume aneinander angeglichen sind. Ohne Angleichung können diese nicht direkt miteinander verglichen werden. Bei unabhängig voneinander erstellten Embeddings ist es möglich, dass die Zeilen in den Vektoren andere Achsen darstellen. Wir untersuchen drei moderne Modelle die dieses Problem auf unterschiedliche Weise angehen.

1) Vector Initialisation (VI): (Kim et al., 2014) Bei VI wird zuerst SGNS auf einem der Korpora trainiert. Dann werden die Gewichte aus dem SGNS Modell gespeichert und genutzt um die Gewichte im zweite SGNS Modell zu initialisieren. Das zweite Modell trainiert dann auf dem zweiten Korpus. Die Intuition zu dieser Methode ist dass die Vektoren für die Wörter schon gelernt sind wenn sie zum Initialisieren genutzt werden. Dadurch sollten sich nur die Vektoren der Wörter ändern welche ihre Bedeutung oder Verwendung geändert haben.

2) Orthogonal Procrustes (OP): (Hamilton et al., 2016a) Hier werden zwei SGNS Modelle unabhängig voneinander auf den beiden Korpora trainiert. Dann wird eine orthogonal beschränkte rotations Matrix berechnet, welche die Vektorräume einander angleicht.

3) Word Injection (WI): (Ferrari et al., 2017) Word Injection fügt hinter alle Ziel Wörter ein spezielles Symbol ein, welches markiert aus welcher Zeit Periode es stammt (t1 oder t2). Anschließend werden die beiden Korpora gemischt und wodurch ein großer Korpus entsteht. Das SGNS Modell welches auf diesem Korpus trainiert wird erzeugt nun für jedes Ziel Wort zwei Vektoren, einen für t1 und einen für t2. Alle restlichen Wörter erhalten jeweils nur einen Vektor. Dadurch dass die Embeddings von selbigen Modell erzeugt wurden sind diese bereits aneinander angeglichen.

Auf Basis von eigenen Beobachtungen während dem Seminar "Lexical Semantic Change Detection" über das Verhalten von VI mit extrem niedriger Dimensionalität und den Werken von Dubossarsky et al. (2018) und Yin und Shan (2018), stellten wir folgende vier Hypothesen auf:

- Die optimale Dimensionalität der drei Methoden ist unterschiedlich.
- Die optimale Dimensionalität von VI ist niedriger als die von OP, und die von OP ist niedriger als die von WI.
- VI hat mehr Störungen als OP, und OP hat mehr als WI.
- Korpus Größe und Epochen Anzahl beeinflussen die optimale Dimensionalität.

Störungen sind definiert als Information, welche nicht semantische Beziehungen zwischen Wörtern beschreiben. Diese Hypothesen dienten als Leitfaden für den ersten Teil der Experimente.

Wir haben die Datensätze von Schlechtweg et. al. 2020 genutzt. Diese bestehen aus vier verschieden Sprachen (Deutsch, Englisch, Latein und Schwedisch), welche unterschiedlichen Korpus Größen haben und aus verschiedenen Zeitperioden stammen. Die deutschen und schwedischen Korpora z.B. sind wesentlich größer als die englischen und lateinischen Korpora. Zu den jeweiligen Sprachen ist auch eine Rangliste gegeben, in der die Ziel Wörter nach ihrem wahren Grad an Bedeutungsänderung aufgelistet sind. Diese Rangliste wurde manuell erstellt und gilt als Referenz wie gut die Modelle Bedeutungswandel erkennen können.

Im ersten Experiment beurteilen wir die Ergebnisse der drei Anpassungs-Methoden mit verschieden Dimensionen. Zusätzlich wird auch eine Strömungsmessung mit verschieden Dimensionen durchgeführt. Mit den Ergebnissen dieses Experimentes versuchen wir die vier Hypothesen zu beantworten. Die erste Hypothese konnte nicht definitiv bestätigt oder widerlegt

werden, da VI und WI oft ähnliche optimale Dimensionalität aufwies. Auch die zweite Hypothese konnte deswegen nicht definitiv beantwortet werden. Dazu kam, dass die optimale Dimensionalität von OP teils über der von VI und WI lag, und teils darunter. Die dritte Hypothese konnte mit dem Vorgehen mit welchem wir in diesem Experiment Störungen gemessen haben, widerlegt werden. Es zeigte sich dass OP häufig die Methode mit den größten Störungen war. VI und WI zeigten wieder ähnliche Werte. Die letzte der vier Hypothesen konnte auch nicht endgültig Beantwortet werden. Wie die Anzahl der Trainings Epochen die optimale Dimensionalität von VI beeinflusst wird in den nachfolgenden Experimenten genauer untersucht. Für OP und WI ließ sich jedoch kein Zusammenhang der beiden Werte erkennen. Wie die Korpusgröße die optimale Dimensionalität beeinflusst konnten wir nicht beantworten, da die Ergebnisse auf den beiden kleineren Korpora sehr große Varianzen aufwiesen und zudem kein klares Optimum der Dimensionalität erkennbar war.

Eine interessante Beobachtung des ersten Experiments ist, dass VI mit höherer Dimensionalität immer schlechtere Ergebnisse erzielt. Das Verhalten sollte jedoch mit größeren Störungen mit hohen Dimensionen erklärt werden können. Es zeigt sich dass, die Vorgehensweise unserer Strömungsmessung die Verschlechterung nicht erklären konnte.

Das nächste Experiment untersucht den Zusammenhang zwischen Wort Frequenz und gemessener Änderung für das entsprechende Wort. Ein solcher Zusammenhang kann als Störung beschrieben werden. Hier zeigt sich bei VI eine Korrelation zwischen Frequenz und der Änderungs Rangliste mit. Die Korrelation wird größer mit zunehmender Dimensionalität. OP und WI zeigen keine signifikante Korrelation der beiden Werte. Bei VI ist zu sehen, dass sobald diese Korrelation einen gewissen Wert überschreitet, die Ergebnisse auf den Testdaten sich beginnen zu verschlechtern. Wir zeigen dass dieser Frequenz BIas die Ursache der schlechten Ergebnisse sind.

Im finalen Experiment werden zwei Vorgehen präsentiert welche den Frequenz Bias stark verhindern und somit Ergebnisse mit hoher Dimensionalität verbessern. Das erste Vorgehen ist erhöhen der Epochen Anzahl und das zweite ist eine längere normalisierung der Vektoren welche benutzt werden um das zweite SGNS Modell zu initialisieren. Warum die jeweiligen Vorgehen den Frequenz Bias verhindern konnte nicht identifiziert werden.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Datum und Unterschrift:

27.07.2020

Declaration

I hereby declare that the work presented in this thesis is entirely my own. I did not use any other sources and references that the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

Date and Signature:

27.07.2020