Time-Out: Temporal Referencing for Robust Modeling of Lexical Semantic Change

Haim Dubossarsky♠ Simon Hengchen♦ Nina Tahmasebi♣ Dominik Schlechtweg♥

♠ Language Technology Lab, University of Cambridge
♦ COMHIS, University of Helsinki
♣ Department of Swedish, University of Gothenburg
♥ Institute for Natural Language Processing, University of Stuttgart

hd423@cam.ac.uk simon.hengchen@helsinki.fi
nina.tahmasebi@gu.se schlecdk@ims.uni-stuttgart.de

Abstract

State-of-the-art models of lexical semantic change detection suffer from noise stemming from vector space alignment. We have empirically tested the Temporal Referencing method for lexical semantic change and show that, by avoiding alignment, it is less affected by this noise. We show that, trained on a diachronic corpus, the skip-gram with negative sampling architecture with temporal referencing outperforms alignment models on a synthetic task as well as a manual testset. We introduce a principled way to simulate lexical semantic change and systematically control for possible biases.

1 Introduction

These past years have seen the rise of computational methods to detect, track, qualify, and quantify how a word’s sense – or senses – change over time. These tasks are critical challenges that are relevant to a range of NLP fields, including the study of historical semantic change. The successful outcome of semantic change detection is relevant to any diachronic textual analysis, including machine translation or normalization of historical texts (Tjong Kim Sang et al., 2017), the detection of cultural semantic shifts (Kutuzov et al., 2017) or applications in digital humanities (Tahmasebi and Risse, 2017a). However, currently, the best-performing models (Hamilton et al., 2016b; Kulkarni et al., 2015; Schlechtweg et al., 2019) require a complex alignment procedure and have been shown to suffer from biases (Dubossarsky et al., 2017). This exposes them to various sources of noise influencing their predictions; a fact which has long gone unnoticed because of the lack of standard evaluation procedures in the field.

We examine the modeling approach of Temporal Referencing (TR) which avoids post hoc alignment and is applicable to any vector space learning technique. We show that it (i) is less affected by noise and (ii) clearly outperforms state-of-the-art alignment models on a synthetic change detection task. The task is based on data from a synchronic corpus into which we artificially inject lexical semantic change (LSC) in a controlled and semantically principled way. We further evaluate the models on a manual testset of diachronic LSC and examine their properties.

In this paper, we focus on skip-gram with negative sampling (SGNS) models (Mikolov et al., 2013) and PPMI (Levy et al., 2015) and make use of TR to share context information across time periods, while learning individual embeddings for a target word in each time period. We evaluate models in two ways: on the one hand, through the comparison of model performance between semantically changing and stable words. This is achieved through the synthetic introduction (and removal) of polysemy, mimicking Schütze (1998); Kulkarni et al. (2015); Rosenfeld and Erk (2018). We differ from previous work by creating those changes in a more structured way, and for many time points. The second type of evaluation put forward is a study built on a smaller number of words manually classified as changed or stable.

Our contributions are the following:

- **Noise Reduction:** We avoid post hoc alignment by TR and show that it outperforms other models and is robust to noise.
- **LSC Simulation:** We propose a systematic and principled method of injecting semantic change in a controlled fashion.
- **Evaluation:** We evaluate (i) by testing for noise reduction in a control condition, (ii) on large and controlled artificial data and (iii) on a manually annotated LSC testset.
- **Framework:** The above comprises a frame-
work to test any model of semantic change for their levels of noise and sensitivity in detecting simulated semantic change.

2 Related Work

Models of LSC Detection Computational approaches to semantic change detection can be divided in different families: count-based semantic spaces (Sagi et al., 2009; Gulordava and Baroni, 2011) and more recently based on neural embeddings (Kim et al., 2014; Basile et al., 2016; Kulka- rni et al., 2015; Hamilton et al., 2016b); graph-based models (Tahmasebi and Risse, 2017a; Mitra et al., 2014, 2015); and finally topic-based (Lau et al., 2012; Wang et al., 2015; Frermann and Lapata, 2016; Hengchen, 2017; Perrone et al., 2019). Recently, we have seen dynamic embeddings with the main aim to circumvent alignment, and share data across time points, thus reducing data volume requirements. Using different base embeddings, SGNS (Bamler and Mandt, 2017), PPMI (Yao et al., 2018), and Bernoulli embeddings (Rudolph and Blei, 2018), the results show that sharing data is beneficial regardless of the method.1 Temporal Referencing has been applied first in the field of term extraction Ferrari et al. (2017) and recently been tested for diachronic LSC detection (Schlechtweg et al., 2019).

Evaluation Due to a lack of proper evaluation methods and datasets, all papers above have performed different, non-comparable evaluations. Previous evaluation procedures mainly tackle a few words: case studies of individual words (Wijaya and Yeniterzi, 2011; Jatowt and Duh, 2014; Hamilton et al., 2016a), or a comparison between a few changing and semantically stable words (Lau et al., 2012; Schlechtweg et al., 2017). Other works focus on the post hoc evaluation of their respective models (Kulkarni et al., 2015; Eger and Mehler, 2016). Importantly, Dubossarsky et al. (2017) proposed to use a control condition to mitigate the absent of validated evaluation methods and datasets.

Control Condition Evaluating empirical results often demands comparing these under a control condition in order to maintain that these are indeed valid and are not the result of unwanted confounding factors. A control condition directly follows from a specific research hypothesis, and therefore must resemble the original condition in any aspect, except the variable of interest that is being hypothesized about. For example, Dubossarsky et al. (2017) attested that a shuffled diachronic corpus is a proper control condition to test models for semantic change, under the hypothesis that such models indeed capture semantic change and not something else. They concluded that any degree of semantic change that is reported by a model on the shuffled corpus may only be related to noise, instead of a true semantic change. Similarly, we propose to test the noise levels associated with different semantic change models using a shuffled historical corpus, and evaluate their true degree of semantic change by comparing their results to the original historical corpus. Importantly, there are many ways to create control conditions, and the synthetic lexical semantic change proposed in Section 4 contains another type of control condition, that is based on artificially induced semantic change.

3 Models

Embeddings A common method in LSC detection is to learn low-dimensional semantic vector spaces (embeddings) for specific time periods and then align spaces for consecutive time periods with an orthogonal mapping which minimizes the distances between the time-specific vectors for all words (Hamilton et al., 2016b). Given two consecutive time periods a, b, and corresponding text corpora C_a, C_b, we learn two vector spaces A, B. Orthogonal Procrustes analysis can then be applied to find the optimal mapping matrix W* such that the sum of squared Euclidean distances between B’s mapping BW and A is minimized:

\[ W^* = \arg \min_W ||BW - A||^2. \]

The optimal solution for this problem is given by an application of Singular Value Decomposition (Artetxe et al., 2017).2 The degree of LSC of a word w is then measured with the cosine distance (Salton and McGill, 1983) between w’s vectors in A and BW* (B’s mapping). This approach

---

1For an extensive survey of computational approaches to lexical semantic change, we refer the readers to Tahmasebi et al. (2018), and to Kutuzov et al. (2018) for a specialized focus on diachronic word embeddings.

2W is constrained to be orthogonal. A and B are first length-normalized and mean-centered and their rows are reduced to the intersection of the vocabulary of C_a and C_b for finding the mapping.
has been found to outperform other LSC detection methods in various studies (Hamilton et al., 2016b; Kulkarni et al., 2015). It has the advantage of not assuming that words keep the same meaning over time. A presumable downside of this approach is expected noise from the alignment, i.e., it may not be possible to align all words to each other that have similar meanings, because the spaces were learned independently.

**PPMI** Another method to learn time-specific semantic vector space representations is to store count-based co-occurrence information for each word in a high-dimensional sparse matrix and then apply Positive Pointwise Mutual Information (PPMI) weighting (Levy et al., 2015). In such a matrix each column stores the co-occurrence statistics with a specific context word. This has the advantage that A and B can be aligned straightforwardly, because many context words occur as columns in both A and B and can hence be mapped onto each other. Mapping A and B to a common coordinate axis then corresponds to intersecting their columns (Hamilton et al., 2016b). This has the advantage of avoiding the complex alignment procedure for embeddings, but also loses their performance advantages (Baroni et al., 2014; Levy et al., 2015).

**Temporal Referencing** Temporal Referencing (TR) is an alternative to learning individual word representations for different time periods, which avoids alignment using a procedure radically simpler than proposed for dynamic embeddings. TR is potentially applicable to every vector space learning method. We treat all time-specific corpora $C_a$, $C_b$, ..., $C_n$ as one corpus $C$ and learn word representations on the full corpus. However, we first replace each target word $w \in C_t$ with a time-specific token $w_t$. This temporal referencing of $w$ is only performed when it is a target word, when the word is considered a context word, it remains unchanged. Following this procedure, we learn one single space that contains a vector for each target-time pair $w_t$, which may be compared directly without the need for alignment. Besides the considerable advantages of avoiding alignment and being applicable to count-based and embedding methods, it presumably lowers data requirements (because context words are collapsed, and thus shared, across corpora). Accordingly, we assume TR to produce smoother change values. As various other models, TR relies on the assumption that the semantics of the context words stays relatively stable over time.

## 4 Synthetic Lexical Semantic Change

We aim to simulate semantic change under controlled settings, while keeping the corpus as natural as possible.\(^4\) We call this procedure sense injection. We increase the semantic material of a recipient word $w^r$ in subsequent subcorpora by injecting contexts from a donor word $w^d$. The context of the recipient word (illustrated as Sense 1 in Figure 1) stays as it is in the corpus. The first subcorpus contains only contexts from the recipient $w^r$ and all the contexts of the donor $w^d$ are removed. In the next time period we add 25% of the contexts of $w^d$, with donor word replaced by the recipient word. In each subsequent corpus, an additional 25% of the donor word are injected until the last time periods contain equal amounts of contexts from the donor and recipient. As a result, seen from the recipient $w^r$, the last time periods have double the amount of contexts as in the first time period $|w^r(t_n) + w^d(t_n)| = 2 * |w^r(t_1)|$.

Note that due to the polysemous nature of words (each is usually associated with more than one sense), we preferred to add the donor words’ contexts instead of simply replacing the existing contexts of the recipient words with the contexts of the donor words. This is because the former involves a single source of synthetic lexical semantic change, while the latter involves two sources (the removal of contexts associated with different senses of a recipient word, as well as the added contexts associated with the senses of a donor word). As a result, this procedure yields less noisy examples of synthetic lexical semantic change.

We differ between cases where recipient and donor are related (e.g. maker $\rightarrow$ creator, Fig. 1a) and unrelated (e.g. shoulders $\rightarrow$ horde, Fig. 1b), following e.g., Pilehvar and Navigli (2013). This procedure is aimed to give us insight into how much novel semantic material is needed for our methods to detect semantic change. Our hypothesis is that cases where the donor word is unrelated to the recipient word should be simpler to detect compared to those that are in close relation. It is

\(^3\)In our case, $t$ is a decade. E.g., in the corpus for 1920 we replace each occurrence of computer with the string computer\(_{1920}\).

\(^4\)Hence, the target words’ frequencies were not matched, but rather stayed natural.
linguistically motivated to choose semantically related words to simulate sense change; those are the most difficult cases of sense change, and a likely procedure of semantic change introducing polysemy (Blank, 1997).

Finally, to simulate the same increase in frequency, we repeat the sense injection for a set of control words. In this case recipient and donor word are the same \( w_r = w_d \). This creates the same increased frequency of the recipient word \( |w_r(t_n)| = 2 \times |w_r(t_1)| \) as the above, but without any added semantic information because the control word keeps its original contexts.

5 Experimental setup

5.1 Corpora

For Experiment 1 (Sec. 6.1) we used COHA (Davies, 2002), of which we restrict ourselves to decadal bins spanning from 1920 to 1970 so as to have a comparable number of tokens for each time slice. For Experiment 2 (Sec. 6.2) we used COCA (Davies, 2008), of which we remove the spoken and academic genres in order to maintain a more similar usage context of words. As a control setting, we created shuffled versions of the same corpora with the same periods, and straightforwardly followed Dubossarsky et al. (2017).

5.2 Synthetic semantic change

For related words, we used the Noun-Noun pairs in SimLex-999 (Hill et al., 2015) as a starting point. However, even semantically unrelated pairs in SimLex were deemed somewhat related by our annotators, and therefore we kept only 10 of those. We created the rest of the list of unrelated words as follows: we randomly sampled 300 lowercased nouns\(^5\) from our corpus, which we assembled into 150 pairs. We then asked three annotators to independently go through the list of generated pairs and determine whether they were semantically related or not. All 150 pairs were deemed semantically unrelated by at least 2 annotators. Only 5 pairs had a disagreement but were qualified as border line cases by the disagreeing annotator, and kept. This procedure yielded 356 word pairs in total, of which 196 were related and 160 were not related.

5.3 Model training

We tested two models in our experiments: (i) low-dimensional embeddings learned with SGNS and (ii) high-dimensional sparse PPMI vectors. Each of these were tested with their respective alignment method (AL) and with Temporal Referencing (TR) as described in Section 3, leaving us with four models to compare:

\[
\begin{array}{cccc}
\text{SGNS}_{\text{AL}} & \text{SGNS}_{\text{TR}} \\
\text{PPMI}_{\text{AL}} & \text{PPMI}_{\text{TR}} \\
\end{array}
\]

In order to avoid that replaced target words co-occur with other target words in TR we used the implementation of Levy et al. (2015), allowing us to train SGNS and PPMI on extracted word-context pairs instead of the corpus directly. For this, we iterated over corpus \( C_t \) such that for each token \( w \) and for each of its context words \( c \) within a symmetric window we extracted the word-context pair: \((w, c)\) if \( w \) is a target word and \((w, c)\) otherwise.

In this way, we guarantee a target word is never replaced and treated as context of any other word. For TR, SGNS and PPMI were then trained on these extracted pairs. For AL, we extracted only regular word-context pairs \((w, c)\) and trained SGNS and PPMI on these. LSC is measured for all four models via cosine distance.\(^6\) (See Appendix A for preprocessing and hyper-parameter details.)

6 Evaluation

To test our methods we performed three main experiments, comparing the performances of TR to the existing state-of-the-art diachronic model

---

\(^5\)The filtering was carried out on the basis of the output of NLTK (Bird et al., 2009)’s `pos_tag` function.

\(^6\)Find a full implementation of the pipeline at [https://github.com/Garrafao/TemporalReferencing](https://github.com/Garrafao/TemporalReferencing).
alignment. In the first experiment, we compare the models’ performance under control conditions that address complementary (potential) weaknesses. The second experiment tests different synthetic change types and assesses whether better models improve detection of lexical semantic change, in a controlled setting. Finally, we test our methods on a manually created testset on a genuine corpus, and manually inspect the results.

6.1 Experiment 1: Model comparison
In this experiment, we trained each model on two corpora, one genuine diachronic corpus with natural semantic change, and one shuffled where the diachronic change is distributed equally across all time periods (see Sec. 5.1). We study the average change of cosine distance as a proxy for semantic change. Following Dubossarsky et al. (2017) we consider the average cosine distance ($\text{acd}_{\text{genuine}}$) trained on the genuine corpus to correspond to true semantic change + noise. In contrast, the average cosine distance on the shuffled corpus corresponds to pure noise. Therefore, the difference between the two equals to true signal, or in other words, true lexical semantic change.

Importantly, we are interested in investigating, and hopefully mitigating, possible sources of the noise that might be found in some of the models. Specifically, we hypothesize that the alignment procedure adds considerable noise to the $\text{acd}$, and plan to test how TR can alleviate some of that noise. Moreover, TR is assumed to contribute not only by circumventing the alignment, but also by producing more stable context vectors due to the increased amount of data on which they are trained.\(^7\) Therefore, we first tease-apart these factors using the following comparisons between the different models.

1. For all models, we consider the difference in average cosine distance between genuine and shuffled conditions ($\text{acd}_{\text{genuine}} - \text{acd}_{\text{shuffled}}$) as being inversely proportional to the amount of noise that the original model unknowingly captures. Hence, the larger the difference, the less noisier (and better) the model is. We consider this to be an approximation of the true semantic change.

2. Focusing on the differences between the two PPMI models allows us to test the independent contribution of TR in providing more accurate context vectors because the intersection of the PPMI vectors are inherently aligned.

3. Focusing on the SGNS models conflates the potential benefits from more accurate context vectors with the disadvantage of Procrustes alignment (which is necessary for SGNS\(_{\text{AL}}\) but not for SGNS\(_{\text{TR}}\)).

4. The difference between the last two would allow us to evaluate the independent contribution of these two sources on the (presumably) less noisy SGNS\(_{\text{TR}}\) model scores.

**Results (experiment 1)** We start analyzing the true semantic change for each of the models (PPMI\(_{\text{AL}}\) to PPMI\(_{\text{TR}}\) and SGNS\(_{\text{AL}}\) to SGNS\(_{\text{TR}}\)) over the corpus. In Figure 2, we can see that temporal referencing introduces less noise throughout the 5 decadal comparisons. For both PPMI and SGNS, the true semantic change increases for the TR models compared to the aligned.

[Figure 2: Comparison of aligned embedding spaces and temporal referencing using both the genuine and the shuffled corpora. High difference in cosine distance indicate less noise captured by the model.]

Importantly, Table 1 shows that for the PPMI models, Temporal Referencing has a much smaller improvement over the aligned model (.005) compared to the SGNS models (.026) (all reported differences are statistically significant, t-test $p < .01$). Temporal Referencing influences the PPMI models only by creating more stable context vectors. In contrast, for the SGNS models the introduction of Temporal Referencing circumvents the use of alignment in addition to creating more stable context vectors. Therefore, the results support our hypothesis that TR has two complementing
factors that improve prior models; firstly, it avoids the need for alignment altogether (and the noise that usually comes with it), and secondly, it produces more stable context vectors due to the increased volume of data when using the full corpus.

Table 1: Difference in average cosine distance between genuine and shuffled conditions (true semantic change) for each method, collapsed over the 5 time bins (1920–1970) in COHA.

<table>
<thead>
<tr>
<th></th>
<th>Align TR</th>
<th>∆</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGNS</td>
<td>0.033</td>
<td>0.059</td>
</tr>
<tr>
<td>PPMI</td>
<td>0.028</td>
<td>0.033</td>
</tr>
</tbody>
</table>

Smoothness of Temporal Referencing  We further analyzed the nature of the progression of the cumulative semantic change that words exhibit over time. Under the assumption that words change their meaning in a systematic way, it follows that words’ semantic change would increase over the years. Therefore, an ecologically valid model of semantic change should show that the words change more as the time interval for comparison increases, for the vocabulary as a whole. In contrast, if a model captures stochastic fluctuations in the words’ vectors instead of true semantic change, then such a shift in the distribution will be less prominent.

We plot the distribution of the words’ cosine distances with increasing time intervals (relative to 1920) for both SGNS models in Figure 3. Both models show a gradual transition from left (smaller change scores) to right (larger change scores). This corroborates our basic assumption that words change more as the time interval for comparison increases. Crucially, Temporal Referencing shows a more constant cumulative progression of cosine distances over time in contrast to alignment where decadal cosine distance distributions seem to be more volatile. We follow Bamler and Mandt (2017) in interpreting these results as attesting for the relatively high noise factor in the SGNS\(_{AL}\) over the SGNS\(_{TR}\).

Overall, the different analyses converge to the same conclusion: Temporal Referencing is a better model for capturing a word’s semantic information from diachronic text because it introduces less noise. Next, we will investigate if a less noisy model is also better at detecting semantic change.

6.2 Experiment 2: Synthetic semantic change

This experiment aims to see how well our methods can find different synthetic change types. In order to minimize natural semantic change in the dataset, we made use of the synchronic dataset COCA which we randomly shuffled, and simulated a diachronic corpus for which we have 7 time-bins. We randomly assigned a seventh of COCA to each of our artificial time periods, labeled \(t_1\) to \(t_7\). Sentences in which either word of the synthetic semantic change pairs (see Sec. 4) or their corresponding control words appeared were held out. These sentences were subsequently added back to COCA according to the procedure outlined in Section 4, which enabled us to control for the fixed ratio incremental steps between the recipient and donor words (i.e., changes to the injection ratio were made only for \(t_2\)-\(t_3\), \(t_3\)-\(t_4\), \(t_4\)-\(t_5\), and \(t_5\)-\(t_6\), while \(t_1\)-\(t_2\) and \(t_6\)-\(t_7\) had no such changes).

All four models were trained on the 7 synthetic time-bins exactly as in Experiment 1. The target words were the 356 words with synthetic lexical semantic change and their 356 control words that were matched with the same frequency increase but otherwise are considered semantically stable. For each target word, the cosine distances between two consecutive synthetic time-bins were computed, resulting in 6 change scores per word.

We analyze the peak distribution of the individual words. We defined the peak position of each word as its vector \(\text{argmax}\) (the position in which
it shows the maximum cosine distance). In order to evaluate the models’ ability to truly detect semantic change, we formulate a naïve binary classification task based on the words’ peak positions. For each word, if the peak is in position 2–5, we classify it as changed, and otherwise as stable and measure accuracy and F1-score.

**Results** Figure 4 shows the $acd$ of the four models for the change and stable words separately, according to the different sense injection ratios. The two plots differ markedly. For the semantic change words (upper plot), all four models show a noticeable peak when the new sense was first injected (step 2), followed by a steady decrease in $acd$ until step 6. In contrast, the stable words only show the steady decrease starting from step 1, without any noticeable peaks. This decrease probably stems from the target words’ increased frequency that can lead to more accurate word embeddings (Hellrich and Hahn, 2016). Because peaks in $acd$ are interpreted as points were semantic change was the most profound, the results support the models’ ability to detect synthetic semantic changes.

![Figure 4: $acd$ at different sense injection steps for the four models. Steps without sense injection are shaded.](image)

Table 2 reports accuracy and F-scores for the four models in the binary classification task. As clearly seen, all four models perform better than chance even under these very rudimentary conditions (finding the $argmax$ of a vector of length 6). Crucially, SGNS$_{TR}$ outperforms the rest of the models, and especially SGNS$_{AL}$ that shows the worst performance. These results corroborate our hypothesis from Experiment 1 that noise is negatively influencing task performance. By alleviating the noise factor that exists in SGNS$_{AL}$ (due to alignment), SGNS$_{TR}$ is able to show substantial gains in this binary classification task.

### Table 2: Accuracy (averaged, and split into individual classes) and F1-scores for semantic change detection.

<table>
<thead>
<tr>
<th></th>
<th>PPMI$_{AL}$</th>
<th>PPMI$_{TR}$</th>
<th>SGNS$_{AL}$</th>
<th>SGNS$_{TR}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stable</td>
<td>0.52</td>
<td>0.54</td>
<td>0.37</td>
<td><strong>0.57</strong></td>
</tr>
<tr>
<td>Unrelated</td>
<td>0.83</td>
<td>0.83</td>
<td>0.86</td>
<td><strong>0.91</strong></td>
</tr>
<tr>
<td>Related</td>
<td>0.73</td>
<td>0.73</td>
<td>0.78</td>
<td><strong>0.78</strong></td>
</tr>
<tr>
<td>Mean acc.</td>
<td>0.65</td>
<td>0.66</td>
<td>0.59</td>
<td><strong>0.70</strong></td>
</tr>
<tr>
<td>F1-score</td>
<td>0.69</td>
<td>0.69</td>
<td>0.67</td>
<td><strong>0.74</strong></td>
</tr>
</tbody>
</table>

**Discussion** Table 2 shows that SGNS$_{TR}$ gains its performance advantage over SGNS$_{AL}$ mainly from a better classification of the stable words (0.37 vs. 0.57). In order to understand this better, we inspect their mean cosine distance curves only for stable words in Figure 5. SGNS$_{TR}$’s curve clearly declines, while SGNS$_{AL}$’s curve declines much less and is more volatile. We attribute the decline of both curves to the diminishing noise that comes from the continuous increase in frequency of the control words (Dubossarsky et al., 2017). It seems that this diminishing frequency noise is counteracted by the alignment noise, yielding a flatter curve for SGNS$_{AL}$. The latter increases SGNS$_{AL}$’s chance to have peaks in one of the center injection steps producing false positives in our classification task. However, this property may also have a positive influence on SGNS$_{AL}$ in related LSC detection tasks (Schlechtweg et al., 2019).

---

8We also ran experiments with moving the time point when the first change was injected and the results mimic those presented here.
6.3 Experiment 3: WSC testset

So far, the results have been based on either a large random sample to show general tendencies for the language in the corpus as a whole, or synthetically injected semantic change. In this part, we test the behavior of our methods on a small, manually created testset for semantic change. We use the Word Sense Change Testset (Tahmasebi and Risse, 2017b) that consists of words and the different associated change events, for the time span 1785 – 2010. In this experiment, we ignore the sense changes and consider only words as changed or stable, and restrict our change words to those that have change events between 1920 and 1970. In total we have 13 changed and 19 stable words (excluding words with a total frequency $\leq 100$).

Table 3: $\text{acd}$ for WSC testset. Var $\in (0.0 – 0.01)$. CH = changed word, ST = stable word, DIFF = difference between ACD for change and stable in percent.

<table>
<thead>
<tr>
<th></th>
<th>SGNS Align</th>
<th>TR</th>
<th>PPMI Align</th>
<th>TR</th>
</tr>
</thead>
<tbody>
<tr>
<td>CH</td>
<td>0.47</td>
<td>0.31</td>
<td>0.86</td>
<td>0.86</td>
</tr>
<tr>
<td>ST</td>
<td>0.34</td>
<td>0.21</td>
<td>0.71</td>
<td>0.73</td>
</tr>
<tr>
<td>DIFF</td>
<td>38%</td>
<td>50%</td>
<td>20%</td>
<td>17%</td>
</tr>
</tbody>
</table>

In Table 3 we see $\text{acd}$ of each model on the changed and stable words. We find that for all methods, SGNS$_{AL}$, SGNS$_{TR}$, PPMI$_{AL}$ and PPMI$_{TR}$, the $\text{acd}$ for the changed words is statistically significantly higher ($p$ values $\leq 0.01$) than for the stable words which nicely corresponds to intuition; words with true semantic change should have vectors that differ more than words without change. The mean difference between the stable and the changed words, that gives us some notion of how well the two different classes are separated, is highest for SGNS$_{TR}$. Because of the limited size of the testset, the results are indicative rather than conclusive and we continue with a manual analysis of the nearest neighboring words.

We carry out a qualitative evaluation for the closest neighbors for $\text{computer}$. Upper part SGNS$_{AL}$, lower part SGNS$_{TR}$. A larger rendering of this figure is available in Appendix D.

![Figure 6: Nearest neighbors for computer. Upper part SGNS$_{AL}$, lower part SGNS$_{TR}$. A larger rendering of this figure is available in Appendix D.](image)

of how well the two different classes are separated, is highest for SGNS$_{TR}$. Because of the limited size of the testset, the results are indicative rather than conclusive and we continue with a manual analysis of the nearest neighboring words.

Next, we study a word for which we expect no sense change, namely $\text{ship}$ (see Appendix E). The SGNS$_{AL}$ show a fairly low $\text{acd}$, but still there

---

9 As an example, the word $\text{car}$ is considered stable since its change event occurred before 1920.

10 The closest words in 1920–1940 have high cosine distances and are thus not very related. Still, for each $\text{computer}_{true}$, the other vectors of $\text{computer}$ are among the neighbors, meaning that despite sparsity and little overlap in context, some structure is found.
are large differences in the top neighboring words. The SGNS\textsubscript{TR} show what we expect; the most similar words are the other ship for synthetic change. Var ∈ (0.0 - 0.01). For almost all decades, the most similar words are ship from the decade before and after. The lower words also help describe the meaning of ship, as a boat and later as a spaceship. The pattern of stability is much more clear for SGNS\textsubscript{TR} than SGNS\textsubscript{AL} and holds for most other stable words as well.

For the word tape, that has a change in dominant sense (or an addition of another strong sense) with the addition of the music tape to adhesive tape, we see the same patterns as for ship, but the bottom words contain ribbon, paper, adhesive for 1920–1940 and recorder, recording, stereo in 1950–1970.

For both the real change in Table 3 and the synthetic change in Table 4, we find that SGNS\textsubscript{TR} is best at differentiating between the stable and the change classes for both datasets (50% for WSC and 26% for synthetic change).

7 Conclusions and future work

In this paper, we have empirically tested the temporal referencing method for lexical semantic change. We train one vector space model over the whole corpus, and thus share information of the context words while training individual vectors for each target word and time period. We compare two commonly used models, namely PPMI and SGNS because of their properties; the PPMI model is count-based and does not require alignment across time, while the SGNS model has shown state-of-the-art results in previous work.

We find that the SGNS model trained with Temporal Referencing contains significantly less noise than the standard SGNS for which an alignment is necessary. In comparison, for the PPMI model where no alignment is needed, Temporal Referencing also significantly reduced the noise level, but to a lesser extent.

Next we evaluated whether the noise reduction carries over performance on a synthetic lexical semantic change detection task. We simulated change in a controlled and semantically principled way, using sense injection and showed that words with semantically related and unrelated semantic change can be differentiated from control (stable) words that are not sense injected, but increase in frequency in the same way as the changed words. SGNS with Temporal Referencing outperforms the other methods in correctly classifying the words to the two classes (change vs. stable).

Finally, we evaluated on a small, handcrafted set of change and stable words and found that SGNS with Temporal Referencing gives the largest separation between words that undergo semantic change and those that stay stable over time. In particular, we observe a similar behavior between this smaller testset and the synthetic sense injection, supporting our sense injection method as a good proxy for isolating and studying lexical semantic change.

Our results support the following conclusion; trained on a diachronic corpus, SGNS with Temporal Referencing will capture more true semantic change. In the future, we plan to evaluate Temporal Referencing against the related dynamic embedding models on an annotated empirical lexical change dataset with multiple languages. We also plan on testing how well Temporal Referencing deals with corpora that are too small for alignment-based methods, hopefully opening new avenues of quantitative research.

Acknowledgements

The authors would like to thank Dr. Barbara McGillivray for her encouragement, and the anonymous reviewers for their helpful comments and suggestions. This work has been funded in parts by the University of Helsinki (research visit grant C1/2019 from the Faculty of Arts, to SH), by the project Towards Computational Lexical Semantic Change Detection supported by a project grant (2019–2022; dnr 2018-01184, to NT), the Centre for Digital Humanities at University of Gothenburg, the Konrad Adenauer Foundation and the CRETA center funded by the German Ministry for Education and Research (BMBF), and the Blavatnik Postdoctoral Fellowship.
References


A Pre-processing and Hyperparameter Details

We lower-cased all tokens in the corpora before extracting word-context pairs. For pair extraction we chose a window size of 5 for both, AL and TR. Corpus tokens were skipped as word or context if they did not have a minimum frequency of 100 in the full corpus used (i.e., 1920-1970 for COHA and full COCA) or contained non-alphabetic characters (except hyphens).

We tuned model parameters on the most recent time bin of COHA (2000-2009) based on word similarity task scores (Hill et al., 2015; Finkelstein et al., 2001) reaching near state-of-the-art results (Levy et al., 2015). The parameters for SGNS were \( \text{dim} = 300 \) (vector dimensionality), \( \text{cds} = 0.75 \) (context distribution smoothing), \( k = 5 \) (number of negative samples) and \( \text{ep} = 1 \) (number of training epochs). PPMI was smoothed and shifted Levy et al. (2015). The parameters were \( \text{cds} = 0.75 \) and \( k = 5 \) (shifting parameter).

B Peak distribution analysis

In Figure 7 we present the peak distributions of the four models for the 712 target words (356 changed and 356 stable), color coded according to the true classification (change/stable). The peaks represent the models’ predictions with respect to where the maximal cosine distance is found for each word, which we later use in a naive and rudimentary binary classification task. As can be seen from the different distributions, all models frequently find peaks in position 2 (corresponding to the event of the first sense injection). However, they are still very much different in their overall peak distributions which influence their sensitivity in detecting synthetically semantic changed words (Table 2).

C WSC TestSet

In Table 5 we list the words that have undergone semantic change, as well as the change year(s) and a description of the change. In Table 6 we list words that do not have changed meanings.

D Closest Neighbors for Computer

In Figure 8 we see the closest neighbors for computer, a word we expect to have changed after the invention of the digital computer in the 1940s, for the SGNS aligned version (upper) and SGNS with temporal referencing (lower).
Table 5: Changed words from WSC Testset

<table>
<thead>
<tr>
<th>Word</th>
<th>Change year</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>aeroplane</td>
<td>1919-1920</td>
<td>First use as weapon of war and commercial flights</td>
</tr>
<tr>
<td>cinema</td>
<td>1900</td>
<td>movie theatre</td>
</tr>
<tr>
<td>computer</td>
<td>1940</td>
<td>digital computer</td>
</tr>
<tr>
<td>cool</td>
<td>1964</td>
<td>a way of being</td>
</tr>
<tr>
<td>flight</td>
<td>1918</td>
<td>after WWI commercial aviation grows rapidly</td>
</tr>
<tr>
<td>gay</td>
<td>1985</td>
<td>recommended for use instead of homosexual</td>
</tr>
<tr>
<td>memory</td>
<td>1960</td>
<td>digital memory</td>
</tr>
<tr>
<td>mouse</td>
<td>1965</td>
<td>the computer mouse was introduced</td>
</tr>
<tr>
<td>record</td>
<td>1920</td>
<td>electrical music records</td>
</tr>
<tr>
<td>rock</td>
<td>1950-1960</td>
<td>birth of rock music</td>
</tr>
<tr>
<td>tank</td>
<td>1917</td>
<td>first tank in battle</td>
</tr>
<tr>
<td>tape</td>
<td>1960</td>
<td>common household use of the magnetic tape</td>
</tr>
</tbody>
</table>

Table 6: Stable words from WSC Testset

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>automobile</td>
<td>music</td>
</tr>
<tr>
<td>bank</td>
<td>newspaper</td>
</tr>
<tr>
<td>camera</td>
<td>paper</td>
</tr>
<tr>
<td>car</td>
<td>phone</td>
</tr>
<tr>
<td>deer</td>
<td>ship</td>
</tr>
<tr>
<td>export</td>
<td>symptom</td>
</tr>
<tr>
<td>founder</td>
<td>telephone</td>
</tr>
<tr>
<td>horse</td>
<td>train</td>
</tr>
<tr>
<td>mail</td>
<td>travel</td>
</tr>
<tr>
<td>mirror</td>
<td></td>
</tr>
</tbody>
</table>

E Closest Neighbors for Ship

In Figure 9 we see the closest neighbors for ship, a word we expect to be stable, for the SGNS aligned version (upper) and SGNS with temporal referencing (lower).
Figure 8: Nearest neighbors for computer. Upper part SGNS_{AL}, lower part SGNS_{TR}.

Figure 9: Nearest neighbors for ship. Upper part SGNS_{AL}, lower part SGNS_{TR}.