Explaining and Improving BERT Performance on Lexical Semantic Change Detection

Severin Laicher, Sinan Kurtyigit, Dominik Schlechtweg, Jonas Kuhn and Sabine Schulte im Walde
Institute for Natural Language Processing, University of Stuttgart
{laichesn,kurtyisn,schlecdk,jonas,schulte}@ims.uni-stuttgart.de

Abstract

Type- and token-based embedding architectures are still competing in lexical semantic change detection. The recent success of type-based models in SemEval-2020 Task 1 has raised the question why the success of token-based models on a variety of other NLP tasks does not translate to our field. We investigate the influence of a range of variables on clusterings of BERT vectors and show that its low performance is largely due to orthographic information on the target word, which is encoded even in the higher layers of BERT representations. By reducing the influence of orthography we considerably improve BERT's performance.

1 Introduction

Lexical Semantic Change (LSC) Detection has drawn increasing attention in the past years (Kutuzov et al., 2018; Tahmasebi et al., 2018; Hengchen et al., 2021). Recently, SemEval-2020 Task 1 and the Italian follow-up task DIACR-Ita provided a multi-lingual evaluation framework to compare the variety of proposed model architectures (Schlechtweg et al., 2020; Basile et al., 2020). Both tasks demonstrated that type-based embeddings outperform token-based embeddings. This is surprising given that contextualised token-based approaches have achieved significant improvements over the static type-based approaches in several NLP tasks over the past years (Peters et al., 2018; Devlin et al., 2019).

In this study, we relate model results on LSC detection to results on the word sense disambiguation data set underlying SemEval-2020 Task 1. This allows us to test the performance of different methods more rigorously, and to thoroughly analyze results of clustering-based methods. We investigate the influence of a range of variables on clusterings of BERT vectors and show that its low performance is largely due to orthographic information on the target word which is encoded even in the higher layers of BERT representations. By reducing the influence of orthography on the target word while keeping the rest of the input in its natural form we considerably improve BERT's performance.

2 Related work

Traditional approaches for LSC detection are type-based (Dubossarsky et al., 2019; Schlechtweg et al., 2019). This means that not every word occurrence is considered individually (token-based); instead, a general vector representation that summarizes every occurrence of a word (including polysemous words) is created. The results of SemEval-2020 Task 1 and DIACR-Ita (Basile et al., 2020; Schlechtweg et al., 2020) demonstrated that overall type-based approaches (Asgari et al., 2020; Kaiser et al., 2020; Pražák et al., 2020) achieved better results than token-based approaches (Beck, 2020; Kutuzov and Giulianelli, 2020; Laicher et al., 2020). This is surprising, however, for two main reasons: (i) contextualized token-based approaches have significantly outperformed static type-based approaches in several NLP tasks over the past years (Ethayarajh, 2019). (ii) SemEval-2020 Task 1 and DIACR-Ita both include a subtask on binary change detection that requires to discover small sets of contextualized usages with the same sense. Type-based embeddings do not infer usage-based (or token-based) representations and are therefore not expected to be able to find such sets (Schlechtweg et al., 2020). Yet, they show better performance on binary change detection than clusterings of token-based embeddings (Kutuzov and Giulianelli, 2020).

3 Data and evaluation

We utilize the annotated English, German and Swedish datasets (ENG, GER, SWE) underlying

Proceedings of the 16th Conference of the European Chapter of the Associationfor Computational Linguistics: Student Research Workshop, pages 192–202
April 19 - 23, 2021.

SemEval-2020 Task 1 (Schlechtweg et al., 2020). Each dataset contains a list of target words and a set of usages per target word from two time periods, t_1 and t_2 (Schlechtweg et al., submitted). For each target word, a Word Usage Graph (WUG) was annotated, where nodes represent word usages, and weights on edges represent the (median) semantic relatedness judgment of a pair of usages, as exemplified in (1) and (2) for the target word *plane*.

(1) Von Hassel replied that he had such faith in the **plane** that he had no hesitation about allowing his only son to become a Starfighter pilot.

(2) This point, where the rays pass through the perspective **plane**, is called the seat of their representation.

The final WUGs were clustered with a variation of correlation clustering (Bansal et al., 2004) (see Figure 1 in Appendix A, left) and split into two subgraphs representing nodes from t_1 and t_2 respectively (middle and right). Clusters are interpreted as senses, and changes in clusters over time are interpreted as lexical semantic change. Schlechtweg et al. then infer a binary change value $B(w)$ for Subtask 1 and a graded change value $G(w)$ for Subtask 2 from the two resulting time-specific clusterings for each target word w.

The evaluation of the shared task participants only relied on the change values derived from the annotation, while the annotated usages were not released. We gained access to the data set, which enables us to relate performances in change detection to the underlying data.[1] We can also analyze the inferred clusterings with respect to bias factors, and compare their influence on inferred vs. gold clusterings. A further advantage of having access to the underlying data is that it reflects more accurately the annotated change scores. In SemEval-2020 Task 1 the annotated usages were mixed with additional usages to create the training corpora for the shared task, possibly introducing noise on the derived change scores.

4 Models and Measures

BERT Bidirectional Encoder Representations from Transformers (BERT, Devlin et al., 2019) is a transformer-based neural language model designed to find contextualised representations for text by analysing left and right contexts. The base version processes text in 12 different layers. In each layer, a contextualized token vector representation is created for every word. A layer, or a combination of multiple layers (we use the average), serves as a representation for a token. For every target word, we feed the usages from the SemEval data set into BERT and use the respective pre-trained cased base model to create token embeddings.[2]

Clustering LSC can be detected by clustering the token vectors from t_1 and t_2 into sets of usages with similar meanings, and then comparing these clusters over time (cf. Schütze, 1998; Navigli, 2009). This section introduces the clustering algorithms and clustering performance measures that we used. **Agglomerative Clustering** (AGL) is a hierarchical clustering algorithm starting with each element in an individual cluster. It then repeatedly merges those two clusters whose merging maximizes a predefined criterion. We use Ward's method, where clusters with the lowest loss of information are merged (Ward Jr, 1963). Following Giulianelli et al. (2020) and Martinc et al. (2020a), we estimate the number of clusters k with the **Silhouette Method** (Rousseeuw, 1987): we perform a cluster analysis for each $2 \leq k \leq 10$ and calculate the silhouette index for each k. The number of clusters with the largest index is used for the final clustering. The **Jensen-Shannon Distance** (JSD) measures the difference between two probability distributions (Lin, 1991; Donoso and Sanchez, 2017). We convert two time specific clusterings into probability distribution P and Q and measure their distance $JSD(P, Q)$ to obtain graded change values (Giulianelli et al., 2020; Kutuzov and Giulianelli, 2020). If P and Q are very similar, the JSD returns a value close to 0. If the distributions are very different, the JSD returns a value close to 1. **Spearman's Rank-Order Correlation Coefficient** ρ measures the strength and the direction of the relationship between two variables (Bolboaca and Jäntschi, 2006) by correlating the rank order of two variables. Its values range from -1 to 1, where 1 denotes a perfect positive relationship between the two variables, and -1 a perfect negative relationship. 0 means that the two variables are not related.

[1] We had no access to the Latin annotated data. For the ENG clustering experiments we use the full annotated resource containing three additional graphs (Schlechtweg et al., submitted).

[2] We first clean the GER usages by replacing historical with modern characters.

Cluster bias We perform a detailed analysis on what the inferred clusters actually reflect. We test hypotheses on *word form*, *sentence position*, *number of proper names* and *corpus*. The influence strength of each of these variables on the clusters is measured by the **Adjusted Rand Index** (ARI) (Hubert and Arabie, 1985) between the inferred cluster labels for each test sentence and a labeling for each test sentence derived from the respective variable. For the variable *word form*, we assign the same label to each use where the target word has the same orthographic form (same string). If ARI = 1, then the inferred clusters contain only sentences where the target word has the same form. For *sentence position* each sentence receives label 0, if the target word is one of the first three words of the sentence, 2, if the target word is one of the last three words, else 1.[3] For *proper names* a sentence receives label 0, if no proper names are in the sentence, 1, if one proper name occurs, else 2.[4] The hypothesis that proper names may influence the clustering was suggested in Martinc et al. (2020b). For *corpora*, a sentence is labeled 0, if it occurs in the first target corpus, else 1.

Average measures Given two sets of token vectors V_1 and V_2 from t_1 and t_2, **Average Pairwise Distance** (APD) is calculated by randomly picking n vectors from both sets, calculating their pairwise cosine distances $d(x, y)$ where $x \in V_1$ and $y \in V_2$ and averaging over these. (Schlechtweg et al., 2018; Giulianelli et al., 2020). We determine n as the minimum size of V_1 and V_2. **APD-OLD/NEW** measure the average of pairwise distances within V_1 and V_2, respectively. They are calculated as the average distance of max. $10,000$ randomly sampled unique combinations of vectors from either V_1 or V_2. **COS** is calculated as the cosine distance of the respective mean vectors for V_1 and V_2 (Kutuzov and Giulianelli, 2020).

5 Results

5.1 Clustering

Because of the high computational load, we apply the clustering only to the ENG and the GER part of the SemEval data set. For this, we use BERT to create token vectors and cluster them with AGL, as described above. We then perform a detailed analysis of what the clusters reflect.[5]

We report a subset of the clustering experiment results in Table 1, the complete results are provided in Appendix B. Table 1 shows JSD performance on graded change (ρ), clustering performance (ARI) as well as the ARI scores for the influence factors introduced above, across BERT layers. For each influence factor we add two baselines: (i) The random baseline measures the ARI score of the influence factor using random cluster labels, and (ii) the actual baseline measures the ARI score between the true cluster labels and the influence factor. In other words, (i) and (ii) respectively answer the question of how strong the influence factor is by chance, and how strong it is according to the human annotation. The values of the two baselines are crucial: If an influence factor has an ARI score greater than both baselines, the clustering reflects the influence factor more than expected. If additionally the influence factor has an ARI score greater than the actual performance ARI score, the clustering reflects the partitioning according to the influence factor more than the clustering derived from human annotations.

Word form bias As explained above, the word form influence measures how strongly the inferred clusterings represent the orthographic forms of the target word. Table 1 shows that for both GER and ENG the form bias of the raw token vectors (column 'Token') is extremely high and always yields the highest influence score for each layer combination of BERT. Additionally, the influence of the word form is significantly higher when using lower layers of BERT. This fits well with the observations of Jawahar et al. (2019) that the lower layers of BERT capture surface features, the middle layers capture syntactic features and the higher layers capture semantic features of the text. With the first layer of BERT the sentences are almost exclusively (.9) clustered according to the form of the target word (e.g. plural/singular division). Even in the higher layers word form influence is considerable in both languages (layer 12: $\approx .4$). This strongly overlays the semantic information encoded in the vectors, as we can see in the low ρ and ARI scores, which are negatively correlated with word form

[3] We assume that especially the beginning and ending of a sentence have a strong influence.

[4] The influence of proper names is only measured for ENG, since no POS-tagged data was readily available for GER.

[5] We also run most of our experiments with k-means (Forgy, 1965). Both algorithms performed similarly with a slight advantage for AGL. We therefore only report the results achieved using AGL.

	Layer	Token	Lemma	TokLem
ρ	1	-.141	-.033	.100
	12	.205	.154	.168
	9-12	.325	**.345**	.293
ARI	1	.022	.041	.045
	12	.116	.111	.158
	9-12	.150	.159	**.163**
Form	1	**.907**	.014	.014
	12	**.389**	**.018**	**.077**
	9-12	**.334**	**.018**	**.051**
Position	1	.001	**.026**	**.024**
	12	**.012**	**.012**	**.015**
	9-12	.002	**.007**	**.003**
Corpora	1	**.019**	**.021**	**.033**
	12	**.078**	**.056**	**.082**
	9-12	**.056**	**.044**	**.063**
Names	1	-.007	.010	.010
	12	**.025**	**.027**	**.033**
	9-12	.019	**.022**	**.026**

	Layer	Token	Lemma	TokLem
ρ	1	-.265	-.062	-.170
	12	.123	.427	**.624**
	9-12	.122	.420	.533
ARI	1	.033	.002	.003
	12	.119	.159	**.161**
	9-12	.155	.142	.154
Form	1	**.706**	.024	.004
	12	**.439**	**.056**	**.150**
	9-12	**.420**	**.047**	**.094**
Position	1	.005	**.023**	**.027**
	12	-.002	.005	-.002
	9-12	**.009**	**.018**	**.012**
Corpora	1	.074	.003	.005
	12	**.110**	**.095**	**.096**
	9-12	**.107**	.068	**.089**
Names	1	-	-	-
	12	-	-	-
	9-12	-	-	-

Table 1: Overview of English clustering scores (left) and German clustering scores (right). Bold font indicates best scores for ρ and ARI (top) or scores above all corresponding baselines for influence variables (bottom).

influence.[6]

The word form bias seems to be lower in GER than in ENG (layer 1: .7 vs. .9). However, this is misleading, as our approach to measure word form influence does not capture cases where vectors cluster according to subword forms as in the case of *Ackergerät*. Its word forms differ as to whether they are written with an 'h' or not, as in *Ackergerät* vs. *Ackergeräth*. As a manual inspection shows this is strongly reflected in the inferred clustering. However, these forms then further subdivide into inflected forms such as *Ackergeräthe* and *Ackergeräthes*, which is reflected in our influence variable. For these cases, our approach tends to underestimate the influence of the variable.

In order to reduce the influence of word form we experiment with two pre-processing approaches: (i) We feed BERT with lemmatised sentences (Lemma) instead of raw ones. (ii) We only replace the target word in every sentence with its lemma (TokLem). TokLem is motivated by the fact that BERT is trained on raw text. Thus, we assume that BERT is more familiar with non-lemmatised sentences and therefore expect it to work better on raw text. In order to continue working with non-lemmatised sentences we only remove the target word form bias by exchanging the target word with its lemma.

As we can see in Table 1, lemmatisation strongly reduces the influence of word form, as expected.[7] Accordingly, ρ and ARI improve. However, it also leads to deterioration in some cases. Also, TokLem reduces the influence of word form and in most cases yields the overall maximum performance. The ARI scores for both languages are similar ($\approx$ .160) while the ρ performance varies very strongly between languages, achieving a very high score for GER (.624).

Replacing the target word by its lemma form seems to shift the word form influence in the different layers: Especially for GER, layers 1 and 1+12 show the highest influences (.706 and .687) with Token (see also Appendix B). In combination with TokLem, both layers are influenced the least (.004 and .046). For ENG we see the same effect for layer 1.

Other bias factors We can see in Table 1 that most influences are above-baseline. As explained above, the word form bias heavily decreases using higher layers of BERT. For all other influences the bias increases when using high layers of BERT.

[6]Note that it is very difficult to reach high ARI scores because ARI incorporates chance.

[7]In some cases it is however above the baselines, indicating that word form is correlated with other sentence features.

This may be because decreasing the word form influence reveals the existence of further –less strong but still relevant– influences. The same is observable with the Lemma and TokLem results, since there the form influence is decreased or even eliminated. While for ENG the influence scores mostly increase using Lemma and TokLem, for GER only the position influence increases, while corpora influence decreases. This is probably because the corpora influence is to some extent related to word form, which often reflects time-specific orthography as in *Ackergeräth* vs. *Ackergerät*, where the spelling with the "h" mostly occurs in the old corpus.

Influence of position and proper names seems to be less important but the respective scores are still most of the times higher than the baselines. So overall the reflection of the two corpora seems to be the most influential factor apart from word form. Often the corpus bias is almost as high as the actual ARI score.

5.2 Average Measures

For the average measures we perform experiments for all three languages (ENG, GER, SWE).

Layers Because we observe a strong variation of influence scores with layers, as seen in Section (5.1), we test different layer combinations for the average measures. The following are considered: 1, 12, 1+12, 1+2+3+4 (1-4), 9+10+11+12 (9-12). As shown in Table 2, the choice of the layers strongly affects the performance. We see that for APD the higher layer combinations 12 and 9-12 perform best across all three languages, while the latter is slightly better (.571, .407 and .554). Interestingly, these two are the only layer combinations that do not include layer 1. All three layer combinations that include layer 1 are significantly worse in comparison. While COS performs best with layer combination 1-4 for ENG (.390), for GER and SWE we see a similar trend as with APD. Again, the higher layer combinations perform better than the other three, which all include layer 1. For GER layer combination 12 (.472) performs best, while 9-12 yields the highest result for SWE (.183). Our results are mostly in line with the findings of Kutuzov and Giulianelli (2020) that APD works best on ENG and SWE, while COS yields the best scores for GER.

Pre-processing As with the clustering, we try to improve the performance of the average measures by using the two above-described pre-processing approaches. We perform experiments only for three layer combinations in order to reduce the complexity: (i) 12 and (ii) 9-12 perform best and are therefore obvious choices. (iii) From the remaining combinations 1+12 shows the most stable performance across measures and languages. Table 3 shows the performance of the pre-processings (Lemma, TokLem) over these three combinations. We can see that both APD and COS perform slightly worse for ENG when paired with a pre-processing (exception to this is 1+12 Lemma). In contrast, GER profits heavily: While APD with layer combinations 12 and 9-12 performs slightly worse with Lemma, and slightly better with TokLem, we observe an enormous performance boost for layer combination 1+12 (.643 Lemma and .731 TokLem). We achieve a similar boost for all three layer combinations with COS as a measure. We reach a top performance of .755 for layer 12 with TokLem. SWE does not benefit from Lemma. We observe large performance decreases, with the exception of combination 1+12 (APD). The APD performance of layers 12 and 9-12 is slightly worse with TokLem. However, layers 1+12, which performed poorly without pre-processing, reaches peak performance of .602 with TokLem. All COS performances increase with TokLem, but are still well below the APD counterparts. The general picture is that GER and SWE profit strongly from TokLem.

Layer	APD			COS		
	ENG	GER	SWE	ENG	GER	SWE
1	.297	.205	.228	.246	.246	.089
12	.566	.359	.529	.339	**.472**	.134
1+12	.455	.316	.280	.365	.373	.077
1-4	.431	.227	.355	**.390**	.297	.079
9-12	**.571**	**.407**	**.554**	.365	.446	**.183**

Table 2: Token performance for different layer combinations across languages.

Word form bias In order to better understand the effects of layer combinations and pre-processing, we compute correlations between word form and model predictions. To lessen the complexity, only layer combination 1+12 (which performed worst overall and includes layer 1), layer combination 9-12 (which performed best overall) in combination with Token and the superior TokLem are considered. The results are presented in Table 4. We observe similar findings for all three languages. The correlation between word form and APD pre-

		Layer	Token	Lemma	TokLem
ENG	APD	12	**.566**	.483	.494
		1+12	.455	**.483**	.455
		9-12	**.571**	.493	.547
	COS	12	**.339**	.251	.331
		1+12	**.365**	.239	.193
		9-12	**.365**	.286	.353
GER	APD	12	.359	.303	**.456**
		1+12	.316	.643	**.731**
		9-12	.407	.305	**.516**
	COS	12	.472	.693	**.755**
		1+12	.373	.698	**.729**
		9-12	.446	.689	**.726**
SWE	APD	12	**.529**	.214	.505
		1+12	.280	.368	**.602**
		9-12	**.554**	.218	.531
	COS	12	.134	-.019	**.285**
		1+12	.077	.012	**.082**
		9-12	.183	-.002	**.284**

Table 3: Performance of pre-processing variants for three layer combinations.

dictions is strong (.613, .554 and .730) for layers 1+12 without pre-processing. The correlation is much weaker with layers 9-12 (.068, .292 and .237) or TokLem (−.026, .105 and .176). This is in line with the performance development that also increases using layers 9-12 or TokLem. Both approaches (different layers, pre-processing) result in a considerable performance increase as described previously. Using layer combination 9-12 with TokLem further decreases the correlation (with the exception of ENG). However, the performance is better when only one of these approaches is used. The correlation between word form and COS model predictions is weaker overall (.246, .387 and .429). We see a similar correlation development as for APD, however this time the performance of ENG does not profit from the lowered bias (see Table 3). Both GER and SWE see a performance increase when the word form bias is lowered by either using layers 9-12 or TokLem.

Polysemy bias The SemEval data sets are strongly biased by polysemy, i.e., a perfect model measuring the true synchronic target word polysemy in either t_1 or t_2 could reach above .7 performance (Schlechtweg et al., 2020). We use APD-OLD and APD-NEW (see Section 4) to see whether we can exploit this fact to create a purely synchronic polysemy model with high performance. We achieve moderate performances for ENG and GER (.274/.332 and .321/.450 respectively) and a good performance for SWE (.550/.562). While the performance for ENG and GER is clearly below the high-scores, the performance is high for a measure that lacks any kind of diachronic information. And in the case of SWE, the performance of both APD-OLD and APD-NEW is just barely below the high-scores (cf. Table 3). Note that regular APD (in contrast to COS) is, in theory, affected by polysemy (Schlechtweg et al., 2018). It is thus possible that APD's high performance stems at least partly from this polysemy bias. This is supported by comparing the SWE results of APD and COS in Table 3: COS is weakly influenced by polysemy and performs poorly, while APD has higher performance, but only slightly above the purely synchronic measures APD-OLD/NEW.

		Layer	Token	TokLem
ENG	APD	1+12	.613	-.026
		9-12	.068	.090
	COS	1+12	.246	-.062
		9-12	.020	.004
GER	APD	1+12	.554	.271
		9-12	.292	.105
	COS	1+12	.387	-.017
		9-12	.205	-.008
SWE	APD	1+12	.730	.176
		9-12	.237	.048
	COS	1+12	.429	-.031
		9-12	.277	-.035

Table 4: Correlations of word form and predicted change scores.

6 Conclusion

BERT token representations are influenced by various factors, but most strongly by target word form. Even in higher layers this influence persists. By removing the form bias we were able to considerably improve the performance across languages. Although we reach comparably high performance with clustering for graded change detection in German, average measures still perform better than cluster-based approaches. The reasons for this are still unclear and should be addressed in future research. Furthermore, we used BERT without fine-tuning. It would be interesting to see how fine-tuning interacts with influence variables and whether it further improves performance.

References

Ehsaneddin Asgari, Christoph Ringlstetter, and Hinrich Schütze. 2020. EmbLexChange at SemEval-2020 Task 1: Unsupervised Embedding-based Detection of Lexical Semantic Changes. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for Computational Linguistics.

Nikhil Bansal, Avrim Blum, and Shuchi Chawla. 2004. Correlation clustering. *Machine Learning*, 56(1-3):89–113.

Pierpaolo Basile, Annalina Caputo, Tommaso Caselli, Pierluigi Cassotti, and Rossella Varvara. 2020. Overview of the EVALITA 2020 Diachronic Lexical Semantics (DIACR-Ita) Task. In *Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020)*, Online. CEUR.org.

Christin Beck. 2020. DiaSense at SemEval-2020 Task 1: Modeling sense change via pre-trained BERT embeddings. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for Computational Linguistics.

Sorana-Daniela Bolboaca and Lorentz Jäntschi. 2006. Pearson versus spearman, kendall's tau correlation analysis on structure-activity relationships of biologic active compounds. *Leonardo Journal of Sciences*, 5(9):179–200.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Gonzalo Donoso and David Sanchez. 2017. Dialectometric analysis of language variation in twitter. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 16–25, Valencia, Spain.

Haim Dubossarsky, Simon Hengchen, Nina Tahmasebi, and Dominik Schlechtweg. 2019. 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. Association for Computational Linguistics.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Edward W. Forgy. 1965. Cluster analysis of multivariate data: Efficiency vs. interpretability of classifications. *Biometrics*, 21:768–780.

Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3960–3973, Online. Association for Computational Linguistics.

Simon Hengchen, Nina Tahmasebi, Dominik Schlechtweg, and Haim Dubossarsky. 2021. Challenges for computational lexical semantic change. In Nina Tahmasebi, Lars Borin, Adam Jatowt, Yang Xu, and Simon Hengchen, editors, *Computational Approaches to Semantic Change*, volume Language Variation, chapter 11. Language Science Press, Berlin.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2:193–218.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Jens Kaiser, Dominik Schlechtweg, and Sabine Schulte im Walde. 2020. OP-IMS @ DIACR-Ita: Back to the Roots: SGNS+OP+CD still rocks Semantic Change Detection. In *Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020)*, Online. CEUR.org. Winning Submission!

Andrey Kutuzov and Mario Giulianelli. 2020. UiO-UvA at SemEval-2020 Task 1: Contextualised Embeddings for Lexical Semantic Change Detection. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for Computational Linguistics.

Andrey Kutuzov, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. 2018. Diachronic word embeddings and semantic shifts: a survey. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1384–1397, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Severin Laicher, Gioia Baldissin, Enrique Castaneda, Dominik Schlechtweg, and Sabine Schulte im Walde. 2020. CL-IMS @ DIACR-Ita: Volente o Nolente: BERT does not outperform SGNS on Semantic Change Detection. In *Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020)*, Online. CEUR.org.

Jianhua Lin. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37:145–151.

Matej Martinc, Syrielle Montariol, Elaine Zosa, and Lidia Pivovarova. 2020a. Capturing evolution in word usage: Just add more clusters? In *Companion Proceedings of the Web Conference 2020*, WWW '20, pages 343—-349, New York, NY, USA. Association for Computing Machinery.

Matej Martinc, Syrielle Montariol, Elaine Zosa, and Lidia Pivovarova. 2020b. Discovery Team at SemEval-2020 Task 1: Context-sensitive Embeddings not Always Better Than Static for Semantic Change Detection. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for Computational Linguistics.

Roberto Navigli. 2009. Word sense disambiguation: a survey. *ACM Computing Surveys*, 41(2):1–69.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2227–2237, New Orleans, LA, USA.

Ondřej Pražák, Pavel Přibákň, Stephen Taylor, and Jakub Sido. 2020. UWB at SemEval-2020 Task 1: Lexical Semantic Change Detection. In *Proceedings of the 14th International Workshop on Semantic Evaluation*, Barcelona, Spain. Association for Computational Linguistics.

Peter J Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Dominik Schlechtweg, Anna Hätty, Marco del Tredici, and Sabine Schulte im Walde. 2019. 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. Association for Computational Linguistics.

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

Dominik Schlechtweg, Sabine Schulte im Walde, and Stefanie Eckmann. 2018. Diachronic Usage Relatedness (DURel): A framework for the annotation of lexical semantic change. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 169–174, New Orleans, Louisiana.

Dominik Schlechtweg, Nina Tahmasebi, Simon Hengchen, Haim Dubossarsky, and Barbara McGillivray. submitted. DWUG: A large Resource of Diachronic Word Usage Graphs in Four Languages.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123.

Nina Tahmasebi, Lars Borin, and Adam Jatowt. 2018. Survey of Computational Approaches to Diachronic Conceptual Change. *arXiv e-prints*.

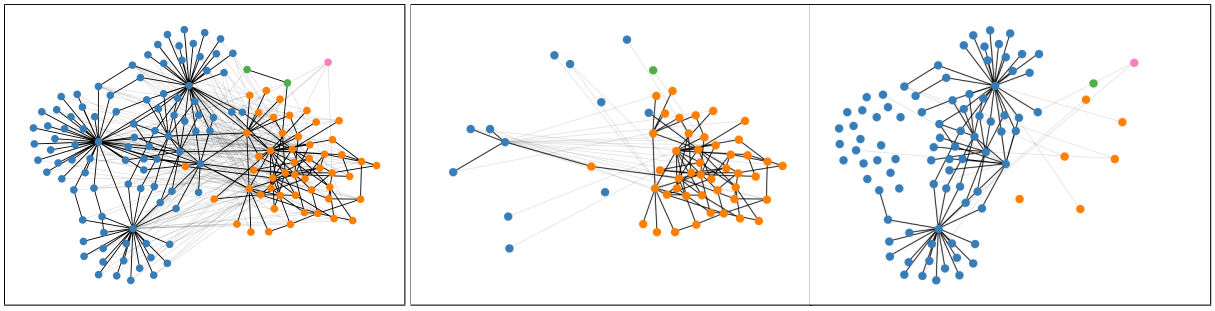
Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244.

A Word Usage Graphs

Please find an example of a Word Usage Graph (WUG) for the German word *Eintagsfliege* in Figure 1 (Schlechtweg et al., 2020, submitted).

B Extended clustering performances and influences

Please find the full results of our cluster experiments in Tables 5 and 6.



full t_1 t_2

Figure 1: Word Usage Graph of German *Eintagsfliege*. Nodes represent uses of the target word. Edge weights represent the median of relatedness judgments between uses (**black**/gray lines for **high**/low edge weights). Colors indicate clusters (senses) inferred from the full graph. $D_1 = (12, 45, 0, 1)$, $D_2 = (85, 6, 1, 1)$, $B(w) = 0$ and $G(w) = 0.66$.

		Layer	Token	Lemma	TokLem
Performance	ρ	1	-.141	-.033	.100
		12	.205	.154	.168
		1+12	-.316	.130	.081
		6+7	.075	-.103	.017
		9-12	.325	**.345**	.293
	ARI	1	.022	.041	.045
		12	.116	.111	.158
		1+12	.022	.141	.149
		6+7	.119	.111	.145
		9-12	.150	.159	**.163**

		Layer	Token	Lemma	TokLem
Performance	ρ	1	-.265	-.062	-.170
		12	.123	.427	**.624**
		1+12	-.252	.235	.401
		6+7	.002	.464	.320
		9-12	.122	.420	.533
	ARI	1	.033	.002	.003
		12	.119	.159	**.161**
		1+12	.037	.064	.080
		6+7	.101	.158	.152
		9-12	.155	.142	.154

Table 5: English clustering performance (left) and German clustering performance (right).

		Layer	Token	Lemma	TokLem
Form	**Influence**	1	.907	.014	.014
		12	.389	.018	.077
		1+12	.881	.020	.057
		6+7	.572	.013	.028
		9-12	.334	.018	.051
	Random	1	.002	.002	.002
		12	-.001	.001	-.001
		1+12	-.002	-.001	-.001
		6+7	.001	.002	.001
		9-12	-.001	-.001	-.002
	Baseline	1	.017	.017	.017
		12	.017	.017	.017
		1+12	.017	.017	.017
		6+7	.017	.017	.017
		9-12	.017	.017	.017
Position	**Influence**	1	.001	.026	.024
		12	.012	.012	.015
		1+12	-.001	.019	.007
		6+7	-.002	.018	-.003
		9-12	.002	.007	.003
	Random	1	.001	.003	.001
		12	.001	-.001	-.001
		1+12	-.001	-.001	-.001
		6+7	.001	-.001	-.001
		9-12	.001	.001	-.001
	Baseline	1	-.002	-.002	-.002
		12	-.002	-.002	-.002
		1+12	-.002	-.002	-.002
		6+7	-.002	-.002	-.002
		9-12	-.002	-.002	-.002
Corpora	**Influence**	1	.019	.021	.033
		12	.078	.056	.082
		1+12	.027	.050	.074
		6+7	.034	.035	.050
		9-12	.056	.044	.063
	Random	1	.001	-.001	.003
		12	.001	.001	.001
		1+12	-.001	.001	.001
		6+7	.001	.001	.002
		9-12	.002	.001	.002
	Baseline	1	.018	.018	.018
		12	.018	.018	.018
		1+12	.018	.018	.018
		6+7	.018	.018	.018
		9-12	.018	.018	.018
Names	**Influence**	1	-.007	.010	.010
		12	.025	.027	.033
		1+12	.018	.022	.027
		6+7	.012	.016	.027
		9-12	.019	.022	.026
	Random	1	-.001	-.002	-.002
		12	-.001	.001	.001
		1+12	-.001	.001	-.001
		6+7	-.001	.001	.001
		9-12	-.001	-.001	.001
	Baseline	1	.019	.019	.019
		12	.019	.019	.019
		1+12	.019	.019	.019
		6+7	.019	.019	.019
		9-12	.019	.019	.019

		Layer	Token	Lemma	TokLem
Form	**Influence**	1	.706	.024	.004
		12	.439	.056	.150
		1+12	.687	.039	.046
		6+7	.503	.050	.050
		9-12	.420	.047	.094
	Random	1	-.001	-.002	.020
		12	-.001	.001	.021
		1+12	-.001	-.001	.020
		6+7	.002	.001	.019
		9-12	.001	-.001	.021
	Baseline	1	.036	.036	.036
		12	.036	.036	.036
		1+12	.036	.036	.036
		6+7	.036	.036	.036
		9-12	.036	.036	.036
Position	**Influence**	1	.005	.023	.027
		12	-.002	.005	-.002
		1+12	.002	.021	.013
		6+7	.010	.020	.018
		9-12	.009	.018	.012
	Random	1	.001	.001	.001
		12	.001	-.001	.001
		1+12	-.001	-.001	.002
		6+7	-.001	.001	.001
		9-12	-.001	.001	.001
	Baseline	1	.005	.005	.005
		12	.005	.005	.005
		1+12	.005	.005	.005
		6+7	.005	.005	.005
		9-12	.005	.005	.005
Corpora	**Influence**	1	.074	.003	.005
		12	.110	.095	.096
		1+12	.077	.024	.052
		6+7	.101	.058	.075
		9-12	.107	.068	.089
	Random	1	-.001	-.001	.001
		12	.001	-.001	.001
		1+12	-.001	.001	.002
		6+7	-.001	.001	-.001
		9-12	-.001	.001	-.001
	Baseline	1	.083	.083	.083
		12	.083	.083	.083
		1+12	.083	.083	.083
		6+7	.083	.083	.083
		9-12	.083	.083	.083
Names	**Influence**	1	-	-	-
		12	-	-	-
		1+12	-	-	-
		6+7	-	-	-
		9-12	-	-	-
	Random	1	-	-	-
		12	-	-	-
		1+12	-	-	-
		6+7	-	-	-
		9-12	-	-	-
	Baseline	1	-	-	-
		12	-	-	-
		1+12	-	-	-
		6+7	-	-	-
		9-12	-	-	-

Table 6: English clustering influences (left) and German clustering influences (right).