Improving Word Usage Graphs with Edge Induction

Bill Noble
University of Gothenburg
bill.noble@gu.se

Francesco Periti
University of Milan
francesco.periti@unimi.it

Nina Tahmasebi
University of Gothenburg
nina.tahmasebi@gu.se

Abstract

This paper investigates *edge induction* as a method for augmenting Word Usage Graphs, in which word usages (nodes) are connected through scores (edges) representing semantic relatedness. Clustering (densely) annotated WUGs can be used as a way to find senses of a word without relying on traditional word sense annotation. However, annotating all or a majority of pairs of usages is typically infeasible, resulting in sparse graphs and, likely, lower quality senses. In this paper, we ask if filling out WUGs with edges *predicted* from the human annotated edges improves the eventual clusters. We experiment with edge induction models that use structural features of the existing sparse graph, as well as those that exploit textual (distributional) features of the usages. We find that in both cases, inducing edges prior to clustering improves correlation with human sense-usage annotation across three different clustering algorithms and languages.

1 Introduction

Recently, Word Usage Graphs (WUGs) have emerged as a new paradigm in the computational study of lexical semantic change (Schlechtweg et al., 2021b). For a given target word (lexeme), a word usage graph consists of a set of *usages*,[1] along with humanly generated *relatedness scores* for some subset of the pairs of usages. Together, the usages (nodes) and relatedness scores (edges) form a weighted graph. Graph clustering techniques can be used to discover word senses, where each cluster of usages is understood to be a distinct sense of the target word.

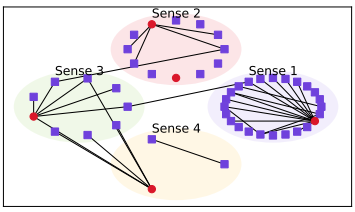
This procedure relies on a simpler human annotation task than assigning a sense from a fixed inventory to each usage, thus allowing us to obtain more annotations with the same number of annotation hours. Moreover, since no sense inventory is required, new or otherwise undocumented senses can be discovered by the procedure. These two factors make WUG annotation particularly useful in applications involving Lexical Semantic Change (LSC), since they make it more feasible to cover a large vocabulary and consider novel or unattested historical senses.

The SemEval-2020 task on Unsupervised Lexical Semantic Change Detection used Diachronic Word Usage Graphs (DWUGs) to develop LSC evaluation datasets for four languages, namely English, German, Swedish, and Latin (Schlechtweg et al., 2020). The use of DWUGs for this purpose has since been adopted in LSC benchmarks for Italian (Basile et al., 2020), Russian (Kutuzov and Pivovarova, 2021), Spanish (Zamora-Reina et al., 2022), Norwegian (Kutuzov et al., 2022), Chinese (Chen et al., 2023), Japanese (Ling et al., 2023), and most recently Slovenian (Pranjić et al., 2024). Each benchmark consists of a diachronic corpus and a set of target words over which human annotation was conducted.

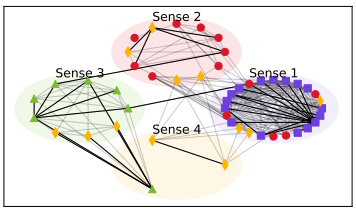
While WUG annotation is less burdensome than traditional word sense annotation, the creation of reliable benchmarks over multiple time periods, still requires a substantial annotation effort. A complete graph on N usages has $(N \cdot (N-1))/2$ edges and, since sense frequency distributions can be highly skewed, sampling a small number of usages does not ensure a representative sample of senses. Thus far, this issue has been addressed by creating simplified LSC benchmarks with reduced annotated edges over two time periods (with the exception of Kutuzov and Pivovarova (2021), who created a benchmark over three time periods). Additionally, as word senses are automatically derived from relatedness judgments of sparse graphs, the evaluation of approaches to LSC is typically conducted through *Graded Change Detection* (i.e., ranking the target words by the degree of semantic change across the corpus) regardless of *Word Sense Induc-*

[1] Contexts drawn from a corpus including the target word.

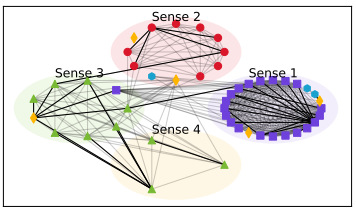
Proceedings of the 5th Workshop on Computational Approaches to Historical Language Change, pages 92–107



(a) Sparse usage relatedness judgments from human annotators (ARI=0.06)



(b) Missing edges inferred with graph-structural features (ARI=0.37)



(c) Missing edges inferred with structural and textual features (ARI=0.62)

Figure 1: The WUGs for *ausspannen*. Only positive (weight ≥ 2.5) edges are shown. Colored regions (labeled Sense 1–4) correspond to human usage-sense annotation, while node colors correspond to clusters found by the `SBM-binomial` model using three different sets of edges: (a) only the human-annotated edges, (b) augmented with induced edges (gray) using *structural evidence*, and (c) augmented edges induced with *structural and textual evidence*. ARI scores indicate correlation with human usage-sense annotation. This example is drawn from Experiment 3 which is described in Section 6.3.

tion (i.e., assessing the quality of word meaning derived by computational models). As a result, more and more so-called *form-based* approaches to LSC have been developed to quantify change. These models sidestep the fundamental aspect of sense modeling that connects LSC to other relevant NLP tasks such as Word Sense Disambiguation and Induction (Periti and Tahmasebi, 2024; Aksenova et al., 2022), and which would make the results of an LSC detection model more interpretable. For example, the SOTA approach to LSC (known as APD) currently consists in measuring the degree of change as average pairwise distance between the contextualized embeddings for a given word (Giulianelli et al., 2020).[2]

In this paper, we investigate *edge induction* as a methodology for augmenting human relatedness judgments in the creation of WUGs, with the goal of reducing the annotation effort required to derive high-quality WUGs. We investigate the following research questions:

RQ1 Can edge induction reduce the human annotation burden required to produce high-quality WUGs?

RQ2 What are the relative contributions of graph-based (structural) and usage-based (contextual) features in WUG edge prediction?

In addition to considering the classification performance of edge induction models, we assess the *quality* of augmented WUGs in terms of how well their node clusters correspond to human-annotated word senses.

[2]We refer the reader to Periti and Montanelli (2024); Tahmasebi et al. (2021); Kutuzov et al. (2018); Tang (2018) for extensive overviews.

2 Related work

In the Word in-Context (WiC) task (Pilehvar and Camacho-Collados, 2019; Loureiro et al., 2022), a model is expected to determine, if two usages of a target word are *related* or *unrelated*. As this is similar to WUG annotation, recent work has shown that large language models such as GPT and BERT can be used as *computational annotators* of DWUGs, reducing the burden of annotation through WiC assessments (Periti and Tahmasebi, 2024; Periti et al., 2024).

This work is similarly motivated. However in contrast to WiC, edge induction leverages a (partial) WUG annotated by humans to infer missing edges, instead of solely relying on models' assessments of relatedness. For example, given a WUG where usage pairs $\langle u, v\rangle$ and $\langle v, w\rangle$ are known to be related, an edge induction model may infer that usages u and w are also related, based on the information provided by the partial graph (see Figure 2a). We use the term *structural features* to denote predictive features derived from the partial graph, and *contextual features* to refer to textual features of the usage applicable in the standard WiC task.

3 Edge induction models

A WUG can be regarded as a weighted graph, with a set of nodes (usages) N, and a weight function $W : E \mapsto \{1, 2, 3, 4\}$,[3] where the domain of W is a subset of pairs of nodes in N; i.e., $E \subseteq \mathcal{E}$, where $\mathcal{E}$ is the set of edges on the complete graph K_N.

[3]These weights correspond to the Likert scale provided to human annotators: *unrelated*, *distantly related*, *closely related*, and *identical*. In some cases, we allow other values in $[1, 4]$, as when the graph is constructed with relatedness scores aggregated over multiple annotator judgments.

An edge induction model is a function that finds a $W' : E' \mapsto \{1, 2, 3, 4\}$ such that $E' \supset E$, while retaining $E' \subseteq \mathcal{E}$. The intended interpretation is that W' *extends* W in such a way that (potentially) uses information encoded in W to induce values $W'(u, v)$ for edges missing from the domain of W.

The simplest operationalization of edge induction is as *classification*, such that, given $\langle u, v\rangle \in \mathcal{E}$, the classifier features can be computed from W (structural features), and potentially some other auxiliary information (as in the case of our textual features). Since our experimental focus is on features, all of the induction models we experiment with in this paper are four-way multi-class logistic regression models provided with different combinations of structural and textual features (described below).

xl-lexeme-cos XL-Lexeme (Cassotti et al., 2023) is an XLM-R-based model trained on a large multilingual corpus of combined WiC datasets. It uses a Siamese architecture similar to sentence BERT (Reimers and Gurevych, 2019), but with the target word marked off by special tokens. The model is trained to minimize the contrasting loss (Hadsell et al., 2006) between pairs of usage embeddings, with cosine distance used as the underlying distance function.

For a pair of usages u and v, let

$$x^{\text{xl-lex}}_{\langle u,v\rangle} = \delta^{\cos}(\mathbf{u}, \mathbf{v}), \tag{1}$$

where $\delta^{\cos}$ is cosine distance and $\mathbf{u}$ and $\mathbf{v}$ are the XL-Lexeme embeddings of usages u and v computed with the lemma of the WUG in question marked as the target.[4]

Since XL-Lexeme is currently state-of-the-art in the WiC task (Periti and Tahmasebi, 2024), we use $x^{\text{xl-lex}}_{\langle u,v\rangle}$ to investigate the predictive contribution of contextual features in our experiments.

log-triangle Intuitively, we should be able to infer something about missing edges based on the edges that have been annotated. This feature works on the intuition of "completing the triangle" between u and v based on the known edges. Suppose we have another usage w and, following Figure 2a, let $x = W(u, w)$ and $z = W(w, v)$ and suppose we know that $x = z = 4$ (i.e,. both pairs $\langle u, w\rangle$ and $\langle w, v\rangle$ are closely related), we might expect that u and v are closely related too and therefore assign $W'(u, v) = \hat{y} := 4$. In fact, given that $x = 4$ we might generalize to expect that $y = z$. However, this is less true when $x = 3$. And when both x and z are 1 or 2, it is difficult to say what can be inferred about y. Moreover, as in Figure 2b, we may have multiple intermediary w_i's that we want to use to "complete the triangle" and aggregate the information provided by their conjunction — pure heuristics won't get us very far.

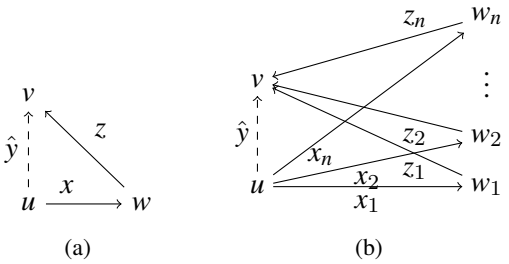


Figure 2: Setup for *triangle path count edge* induction. The value of the missing edge ($\hat{y}$) can be inferred from the weights along each of the known paths $\{(x_1, z_1), ..., (x_n, z_n)\}$ from u to v.

The general case is described by Figure 2. There is no edge between usages u and v, but we do have edges between u and some number of other usages $w_1, w_2, ..., w_n$ and edges between each w_i and v. We define the *triangle path count* as a count vector of the weights along all the length-2 paths from u to v. Formally,

$$\mathbf{x}^{\text{tri}}_{\langle u,v\rangle}[i] = \sum_{w_j} \{1 \mid \langle W(u, w_j), W(w_j, v)\rangle = p_i\}, \tag{2}$$

where p_i indexes the set of the possible length-2 paths of weights (i.e., permutations of $\{1, 2, 3, 4\}$). If all counts $\mathbf{x}^{\text{tri}}_{\langle u,v\rangle}[i] = 0$, then $\mathbf{x}^{\text{tri}}_{\langle u,v\rangle}$ is undefined — assuming that the domain of W is constructed from independently distributed samples from $\binom{N}{2}$, the fact that there is no length-2 path between u and v doesn't tell us anything about what $W'(u, v)$ should be.

To account for the fact that each additional path of a given type likely provides marginally less predictive information about the correct label for $\langle u, v\rangle$, we use the point-wise log of the triangle path count as input features to the logistic regression model.[5]

[4] Note that $x^{\text{xl-lex}}_{\langle u,v\rangle}$ is a scalar value, meaning that the regression model using only this feature essentially finds data-driven thresholds that segment $[-1, 1]$ (the range of $\delta^{\cos}$) into four bins corresponding to the edge annotation schema.

[5] A more natural way to account for this diminishing information content would be with a Multinomial Naive Bayes model, that operates on $\mathbf{x}^{tri}$, however we found the classification performance of that model to be similar to that of the logistic regression model using the log-count feature. For

$$\mathbf{x}^{\text{log-tri}}_{\langle u,v\rangle}[i] = \log(\mathbf{x}^{\text{tri}}_{\langle u,v\rangle}[i] + 1) \quad (3)$$

log-triangle+xl-lexeme-cos Finally, the model that combines textual and and structural features simply uses the concatenation of `xl-lexeme-cos` and the `log-triangle` features:

$$\mathbf{x}^{\text{log-tri+xl-lex}}_{\langle u,v\rangle} = \mathbf{x}^{\text{log-tri}}_{\langle u,v\rangle} \oplus [x^{\text{xl-lex}}_{\langle u,v\rangle}] \quad (4)$$

3.1 Iterated inference

Models that use $\mathbf{x}^{\text{log-tri}}$ (and $\mathbf{x}^{\text{log-tri+xl-lex}}$) have undefined features when there are no length-2 paths from u to v. Suppose we have a trained classifier $\mathcal{C}$ which, given an existing weight function W and auxiliary information A, predicts new weights; i.e., $\mathcal{C}_{W,A} : N \times N \mapsto [1,4]$. Letting W^0 be the initial weight function and $E^0 = \text{Dom}(W^0)$ be the edges for which we have ground-truth weights, we infer edges in stages as follows:

$$W^i(u,v) = \begin{cases} W^0(u,v) & \text{if } \langle u,v\rangle \in E^0 \\ \mathcal{C}_{W^{i-1},A}(u,v) & \text{otherwise.} \end{cases} \quad (5)$$

In other words, we preserve all of the original (ground-truth) edge weights while updating inferred weights with new predictions at each iteration. Other schemes are of course possible, but this one seeks a balance between propagating information from the larger graph at each iteration and remaining grounded in the seed edges (hopefully avoiding excessive error propagation).

3.2 Levels of stratification

There are several choices for how to divide the predictive domain of each classifier. Intuitively, we would expect words to behave similarly with respect to the inferential evidence provided by the $\mathbf{x}^{\text{log-tri}}$ and $x^{\text{xl-lex}}$ features.

But there might be differences across words (especially considering that different words have different patterns of polysemy and part-of-speech) and across languages. Given a limited annotation budget, it would be beneficial to share training data as much as possible. We experiment with three schemes:

this reason and because the logistic regression model is more readily compatible with additional features, we only report the results of the logistic regression models.

word-level A classifier is trained based on the training edges for each word, regardless of language. At inference time, edges are inferred using the word-specific classifier with the same *seen* edges initializing the graph.

language-level Training data is merged across words in a given language. At inference time, the language-specific classifiers are used to predict edges for words in the corresponding language. Graphs are initialized with the word-specific *seen* edges, which may be the edges from the training set (as in Section 6.1) or edges from new words from the same language that weren't seen at train time (as in Section 6.3).

cross-lingual Only one classifier is trained using data from all training words. As before, the classifier can be used to infer edges in WUGs for words both inside or outside of the training set.

3.3 Evaluation: Correlation with human annotators

We evaluate edge induction models by their weighted average pairwise Spearman correlation with human annotators, defined as follows:

$$\frac{\sum_{h\in H} \rho(y_h, \hat{y}_{m,h})}{\sum_{h\in H} \mid y_h \mid}, \quad (6)$$

where y_h is the sequence judgments by annotator h, $\hat{y}_h$ is the corresponding sequence of model predictions on the same items, and ρ is the Spearman correlation coefficient.

Pairwise Spearman correlation is a common metric for evaluating agreement among annotators of usage relatedness (e.g., Erk et al., 2013; Schlechtweg et al., 2021b). We use this metric to evaluate our edge induction models in order to assess how well they perform as computational annotators.

4 Clustering

Correlation Clustering has traditionally been used with sparsely human-annotated WUGs. For example, to identify senses forming the basis of the SemEval-2020 bencmark (Schlechtweg et al., 2020). We also experiment with two varieties of *Stochastic Block Model* (SBM; Holland et al., 1983), a family of generative models which may better accommodate the uncertainty introduced by computationally-annotated edges.

In an SBM, an edge between nodes u and v is determined by a random variable which depends

on the blocks that u and v belong to. The parameters of the distributions that generate edges between pairs of blocks and the block membership of nodes can be jointly inferred through Bayesian non-parametric inference. In this way, SBMs can discover both *assortative* block structures (clusters), in which nodes belonging to the same block are more likely to have an edge, as well as other more general relationships between blocks, as expressed though the graph's edges.

The Hierarchical SBM (Peixoto, 2014) generalizes the SBM by imposing an additional block structure on the first-order blocks. The inferred relationship between — and membership in — second-order blocks allows the model to find informative priors for the first-order blocks. In principle the model can be nested to an arbitrary depth. In practice Peixoto (2014) provides methods to infer the hierarchical structure.[6] One benefit of using hierarchical models is they they can find smaller well-defined blocks compared to vanilla SBM. This is particularly advantageous to our use-case, since sense distributions are known to be highly skewed (Kilgarriff, 2004).

In both standard and hierarchical SBM, block membership determines the likelihood of an edge between two nodes. Unknown values aside, WUGs are complete graphs — the *existence* of an edge is not informative for finding a good clustering. For that reason, we experiment with two SBM variants that can be adapted to our situation.

sbm-binomial The weighted SBM (Aicher et al., 2015; Peixoto, 2018) draws edge weights from a distribution in the exponential family. As with the SBM, these distributions are parametrized by the block membership of the nodes. Schlechtweg et al. (2021a) found the Binomial distribution to have the best fit to WUG edge weights.

sbm-layers We also experiment with an approach that uses the layered model of Peixoto (2015). In this model, each of the four edge weights are treated as a different *type* of edge. The generative process allows the the edge likelihood between blocks to be treated independently for each block while the blocks (clusters) themselves are inferred jointly.

In all of our experiments that use SBM models, we cluster according to the most frequently-assigned blocks over 10 000 samples from the agglomerative Markov chain Monte Carlo algorithm, after first minimizing the entropy of the model.[7]

correlation Correlation clustering (Bansal et al., 2004) scores possible partitions according to the difference between the sum of positive edges *across* clusters and sub of the weight of negative edges *within* clusters. Following (Schlechtweg et al., 2021b), we shift all of the edge weights by 2.5 so that edges weighted 1 and 2 are negative and edges weighted 3 and 4 are positive. We also use their implementation of the cluster search, which uses simulated annealing to approximate an optimal solution.

4.1 Evaluation: Adjusted Rand Index

For two partitions of the same set, the Rand Index (RI; Rand, 1971) measures the proportion of pairs of items that either appear in the same or different clusters in both partitions. RI is a measure of correlation between partitions that, crucially, doesn't rely on any explicit alignment of clusters. The Adjusted Rand Index (ARI; Hubert and Arabie, 1985) accounts for the possibility that pairs of items are assigned together or apart at random by normalizing with the expected value of the RI.

5 Data

Our experiments draw on two sets of WUGs. We use the German `DWUG_DE` dataset (Schlechtweg et al., 2022, v2.3.0). In particular, we use the subset of this data which is additionally annotated with usage-sense annotations (24 of 50 lemmas and 50 of 200 usages per lemma). In contrast to the usage-usage annotation used to construct the WUGs, the usage-sense annotation (Schlechtweg, 2023) was carried out in the traditional way where annotators select a sense from a predefined list of senses. This will allow to evaluate how well the derived WUG clusters correlate with traditional sense annotation. Each usage was annotated by 3 annotators and we use the sense annotated by a majority (2) as the ground-truth. Usages where the annotators disagree (83 out of 1200) are excluded from the correlation analysis.

Additionally, the `resampled` dataset is a larger dataset of WUGs (Schlechtweg et al., 2024)[8] from three languages (German, English, and Swedish),

[6] Both of our SBM models use hierarchical implementations from `graph-tool.skewed.de` (v.2.45).

[7] We use `minimize_nested_blockmodel_dl` with default parameters.

[8] https://www.ims.uni-stuttgart.de/data/wugs

which are much more densely annotated with usage-usage edges. This allows us to experiment with the effect of different amounts of ground-truth data (Sections 6.1 and 6.2).

In Section 6.3, we use the German portion of the DWUG_DE corpus that *doesn't* overlap with the sense-annotated lemmas to test the usefulness of edge induction in a simulated low-data scenario.

Some of the data contains overlapping human usage-usage annotation. In all of our experiments, we use the median (rounded up to the nearest integer) of these judgments as the ground-truth edge scores for clustering and training edge induction models. For testing the edge induction models, we use the disaggregated judgments to compute annotator-wise correlations of the model prediction with human judgments (see Section 3.3).

6 Experiments

Given limited human usage-sense annotation, we conduct two stages of experiments. First (Section 6.1), we use the densely annotated `resampled` WUGs to test how well edge induction models recover edge weights given different amounts of usage-usage annotation for training and graph initialisation. Likewise (Section 6.2), we test the robustness of different clustering methods with respect to recovering sense clusters with limited usage-usage annotated data. Next (Section 6.3) we construct a more realistic scenario in which pre-trained edge induction models are used to predict edges in sparsely-annotated WUGs of "new" words. These enriched graphs are then clustered and compared to human usage-sense annotations.

Ultimately the end-to-end results (correlation with human sense annotation) are what matter, but considering the intermediate results will allow us to better explain the final performance and make recommendations that generalize to more WUG creation scenarios (for example, given different annotation budgets).

6.1 Experiment 1: Edge induction performance

For 5 different folds, we reserve 10% of the edges in each `resampled` WUG for testing. Of the remaining edges, for each fold, we train classifiers with different amounts of training data, from 50 to 300 annotated edges, using each of the stratification schemes described in Section 3.2. At inference time, we initialize the graphs with the edges that were seen during training and infer the remaining edges, including the edges in the respective test set. For models that use $\mathbf{x}^{\text{tri}}$ and $\mathbf{x}^{\text{tri+xl-lex}}$, four rounds of inference are performed.

The results are shown in Figure 3. Overall, the results are good. In the best cases, our models roughly achieve parity with human-human agreement for a moderate number ground-truth of edges (see (Schlechtweg et al., 2021b)), and have decent agreement in low-data scenarios.

The models that combine textual and structural features (i.e., $\mathbf{x}^{\text{tri+xl-lex}}$) perform best for all but the smallest number of ground-truth edges, especially in the language-level and cross-lingual case. It's important to consider that the number of ground-truth edges are reported *per word*, so at 50 ground-truth edges, the cross-lingual model has many more training examples than any of the individual word-level models. However, this is exactly the point of training models at higher levels of stratification, since it makes quality inference more efficient in terms of annotation effort.

We also see that iterated inference does make a difference. For word-level models the performance actually degrades at higher inference interactions, suggesting that the model may suffer from some degree of error propagation. This is not the case with language-level and cross-lingual models, which have richer training sets: subsequent inference iterations do improve the performance, though there is not much change after the second round for the combined model. Crucially, subsequent rounds also have better predictive *coverage*, since the triangle count-based models are unable to make an edge prediction when there is no length-2 path between the corresponding nodes. For the purposes of the clustering, this means that later inference rounds should almost always be preferred, especially in the language-level and cross-lingual setups.

6.2 Experiment 2: Clustering robustness

In this experiment, we use the same data, folds, and training limits as in Experiment 1, this time experimenting with clustering results. The goal of this experiment is to observe the stability of each clustering algorithm given different numbers of ground-truth edges. We perform this experiment as a precursor to clustering on induced edges, since it will provide context for any clustering improvements stemming from edge induction. Each of the algorithms we experiment with is designed to work on graphs with missing edges, so it is important to

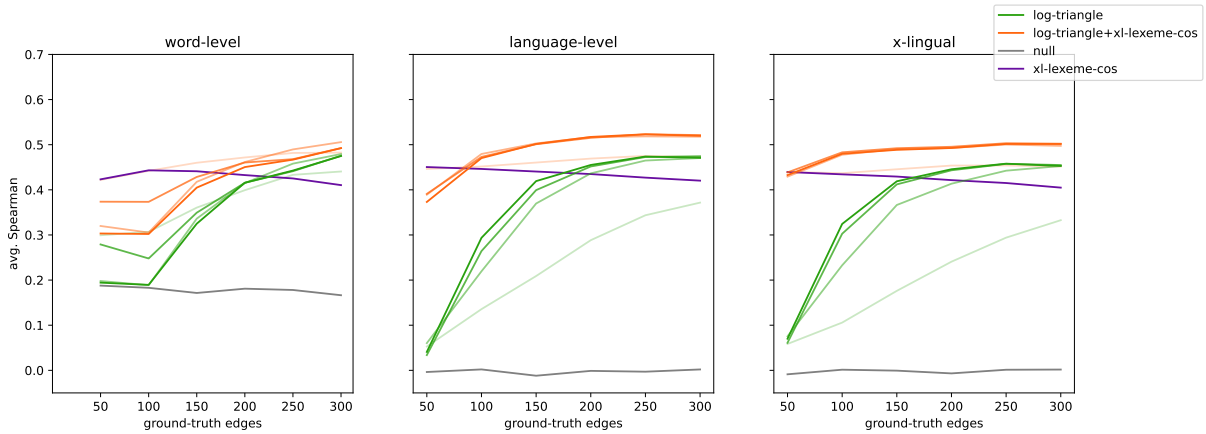


Figure 3: Weighted average Spearman correlations between model predictions and human annotations (see Section 3.3). Here, the scores are computed by considering annotations from all lemmas together, and then averaged over 5 folds. For models using `log-triangle` features, inference iterations are shown with increasingly saturated lines, with iteration 4 being the most saturated. A proportional random baseline (gray) is show for comparison. Analogous to the models, label proportions are computed at the word, language and cross-lingual level. Language-specific results are provided in Appendix A.

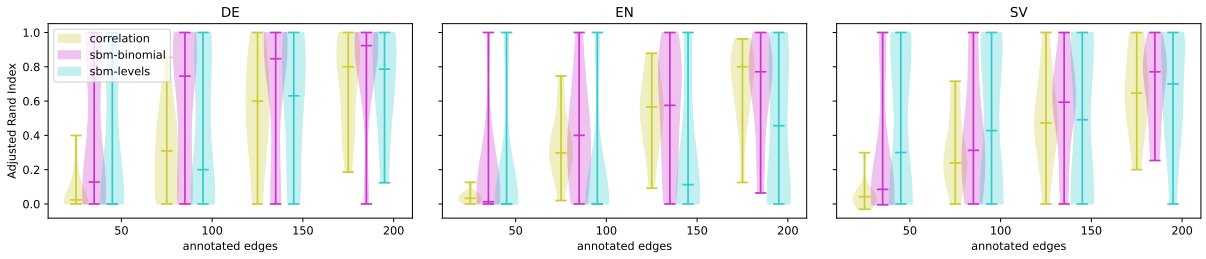


Figure 4: ARI spread over lemmas (first averaged over 5 folds), for different number of edge annotations. For each algorithm the ARI is computed with respect to the clusters achieved by the same clustering algorithm provided with 300 annotations These results are only useful to compare across algorithms insofar as they give an idea of how quickly the algorithm converges to a clustering result.

understand how much the results change for WUGs with different amounts of missing data. For each algorithm, we compute the ARI between clusters produced with 50 to 200 ground-truth edges and clusters produced by the same algorithm with 300 ground-truth edges.

The results are presented in Figure 4. Naturally, all methods produce clusters more similar to the 300-edge clusters when provided with higher numbers of ground-truth edges. With 50 ground-truth edges, clustering is very poor across all clustering algorithms for the majority of words. The results using the `sbm-binomial` method improve fastest with increasing numbers of ground-truth edges, and at 200 edges, it performs best on German and Swedish, while `correlation` performs best for English.

Even at the highest number edges, though, there is a wide spread of performance across words. It is important to interpret all of these results bearing in mind that the ARI is compared to clusters produced by the same algorithm, just with more data. For an extrinsic validation of the clusters, we must turn to Experiment 3, which compares clusters to human-annotated sense data.

6.3 Experiment 3: Realistic scenario

Experiment 3 imitates a scenario in which one has (1) a number of "new" words with very limited edge annotation, and (2) another collection of words with larger and more densely annotated WUGs.

For the sparsely-annotated data, we draw from `DWUG DE`, selecting the 24 words that have been annotated with sense data. For each word, we construct graphs with only the 50 usages that were annotated with sense data and all of the annotated edges that include only those usages (median 55 edges per word; see Appendix B for word-level

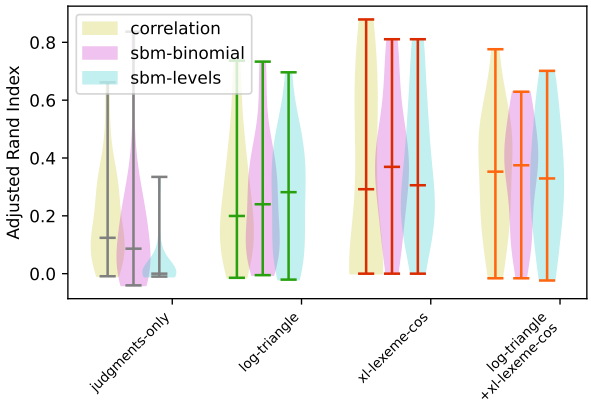


Figure 5: ARI of WUG clusters versus human sense annotation. Spreads are shown over lemmas (N=24). The judgments-only column shows clustering of the graphs based on human usage-usage annotation alone, while the other columns show the effect of adding predicted edges. Disaggregated results for the $\mathbf{x}^{\text{tri+xl-lex}}$ models can be found in Appendix C.

counts) The densely-annotated data, is drawn from the full set of usages of lemmas in DWUG_DE that *don't* overlap with the words annotated for sense (26 words). We reserve 10% of the edges for testing and train language-level classifiers on the remainder.

We then predict edges on the sparsely-annotated WUGs and compare the clusters to human-annotated sense data. As a baseline, we compute clusters for the graphs with only ground-truth edges. For each edge induction model, we compute clusters using the graph enriched with predicted edges (retaining the ground-truth edges that exist).

The results (Figure 5) show clear improvements over the sparse graph clusters for all induction models and clustering algorithms. The `sbm-binomial` algorithm performs slightly better than the `correlation` clustering algorithm on graphs with edges induced by models that include the $x^{\text{xl-lex}}$. Moreover, there is a tighter spread in performance across words for the `sbm-binomial` algorithm.

In cases where $x^{\text{xl-lex}}$ isn't used, `sbm-layers` performs best on average.

In all cases, there are still some words where the correlation with human sense annotation is very poor, median but performance can be improved greatly by using induced edges.

7 Conclusion

In this paper we investigate the question of whether missing edges in WUGs can be induced using information derived from the existing human annotated edges. Our final goal is to improve downstream clustering performance by using only as much human annotation as is needed. To set the stage, we first explore how well edge induction models that exploit structural and textual features correlate with human WUG annotation for different amounts of ground-truth data (Section 6.1). Then, we characterize the stability of clustering algorithms, finding notable differences in the clusters as more ground-truth edges are added across all 3 algorithms (Section 6.2). Finally, we conduct an experiment showing that edge induction models can be used to improve the clustering of sparse WUGs even when they are trained on data from a completely disjoint set of lemmas (Section 6.3). These results show that edge induction can be a valuable tool for improving the quality of sense clusters inferred from sparsely-annotated WUGs. This can allow researchers and lexicographers to cover a larger set of lemmas on a limited annotation budget. It also points to annotation strategies that strategically use triangles to maximize the utility of each human annotation.

Importantly, we saw that both structural (graph-based) and contextual (language model-based) features contribute to the WUG quality improvements resulting from augmenting with induced edges. This is significant since there may be situations when it is desirable to avoid the possibility of introducing historical biases with language model-derived features.

This work leaves room for further improvements on edge induction and clustering in WUGs. The iterated inference strategy described in Section 3.1 is just one of many possible strategies for incorporating distant graph information while minimizing error propagation. More principled approaches, such as Message Passing Neural Networks (Gilmer et al., 2017; Zhang and Chen, 2018) should also be investigated. Likewise, some versions of the Stochastic Block Model (i.e., Peixoto, 2019) can account for missing edges, which theoretically makes joint induction of edges and clusters possible, though no implementation currently exists for weighted networks.

8 Limitations

A notable limitation of the results in Section 6.3 stems from the use of usage-sense annotation for evaluation. One of the motivations for WUGs is that they can be used to discover unattested word senses. By its nature, usage-sense annotation assumes a fixed sense inventory — it could simply be that some of the senses discovered by the clustering process were not present in the sense inventory used for annotation, either because they were missing or because the clusters capture a more fine-grained notion of sense. Nevertheless correlation with usage-sense annotation is an important way to validate that usage clusters correspond to what we think of as word senses.

Finally, in this work, our investigation is confined to English, Swedish, and German WUGs. Since these languages are all closely related, the cross-lingual results should be interpreted with that in mind. Otherwise, our proposed methods are language-agnostic, and we do not anticipate significant challenges in adapting them to other languages.

Acknowledgments

This work has in part been funded by the research program Change is Key! supported by Riksbankens Jubileumsfond (under reference number M21-0021). The computational resources were provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

References

Christopher Aicher, Abigail Z. Jacobs, and Aaron Clauset. 2015. Learning latent block structure in weighted networks. *Journal of Complex Networks*, 3(2):221–248.

Anna Aksenova, Ekaterina Gavrishina, Elisei Rykov, and Andrey Kutuzov. 2022. RuDSI: Graph-based word sense induction dataset for Russian. In *Proceedings of TextGraphs-16: Graph-based Methods for Natural Language Processing*, pages 77–88, Gyeongju, Republic of Korea. Association for Computational Linguistics.

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

Pierpaolo Basile, Annalina Caputo, Tommaso Caselli, Pierluigi Cassotti, and Rossella Varvara. 2020. DIACR-Ita@ EVALITA2020: Overview of the EVALITA2020 DiachronicLexical Semantics (DIACR-Ita) Task. In *Proceedings of the Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA)*, Online. CEUR-WS.

Pierluigi Cassotti, Lucia Siciliani, Marco DeGemmis, Giovanni Semeraro, and Pierpaolo Basile. 2023. XL-LEXEME: WiC pretrained model for cross-lingual LEXical sEMantic changE. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1577–1585, Toronto, Canada. Association for Computational Linguistics.

Jing Chen, Emmanuele Chersoni, Dominik Schlechtweg, Jelena Prokic, and Chu-Ren Huang. 2023. ChiWUG: A Graph-based Evaluation Dataset for Chinese Lexical Semantic Change Detection. In *Proceedings of the 4th Workshop on Computational Approaches to Historical Language Change*, pages 93–99, Singapore. Association for Computational Linguistics.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. 2013. Measuring word meaning in context. *Computational Linguistics*, 39(3):511–554.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. 2017. Neural Message Passing for Quantum Chemistry. *arXiv:1704.01212 [cs]*.

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.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality Reduction by Learning an Invariant Mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA*, pages 1735–1742. IEEE Computer Society.

Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. 1983. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2(1):193–218.

Adam Kilgarriff. 2004. How Dominant Is the Commonest Sense of a Word? In *Text, Speech and Dialogue*, Lecture Notes in Computer Science, pages 103–111, Berlin, Heidelberg. Springer.

Andrey Kutuzov, Lilja Ovrelid, 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.

Andrey Kutuzov and Lidia Pivovarova. 2021. Three-part Diachronic Semantic Change Dataset for Russian. In *Proceedings of the 2nd International Workshop on Computational Approaches to Historical Language Change 2021*, pages 7–13, Online. Association for Computational Linguistics.

Andrey Kutuzov, Samia Touileb, Petter Mæhlum, Tita Enstad, and Alexandra Wittemann. 2022. NorDiaChange: Diachronic Semantic Change Dataset for Norwegian. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2563–2572, Marseille, France. European Language Resources Association.

Zhidong Ling, Taichi Aida, Teruaki Oka, and Mamoru Komachi. 2023. Construction of Evaluation Dataset for Japanese Lexical Semantic Change Detection. In *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, pages 125–136, Hong Kong, China. Association for Computational Linguistics.

Daniel Loureiro, Aminette D'Souza, Areej Nasser Muhajab, Isabella A. White, Gabriel Wong, Luis Espinosa-Anke, Leonardo Neves, Francesco Barbieri, and Jose Camacho-Collados. 2022. TempoWiC: An Evaluation Benchmark for Detecting Meaning Shift in Social Media. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3353–3359, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Tiago P. Peixoto. 2014. Hierarchical Block Structures and High-Resolution Model Selection in Large Networks. *Physical Review X*, 4(1):011047.

Tiago P. Peixoto. 2015. Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. *Physical Review E*, 92(4):042807.

Tiago P. Peixoto. 2018. Nonparametric weighted stochastic block models. *Physical Review E*, 97(1):012306.

Tiago P. Peixoto. 2019. Network Reconstruction and Community Detection from Dynamics. *Physical Review Letters*, 123(12):128301.

Francesco Periti, Haim Dubossarsky, and Nina Tahmasebi. 2024. (Chat)GPT v BERT Dawn of Justice for Semantic Change Detection. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 420–436, St. Julian's, Malta. Association for Computational Linguistics.

Francesco Periti and Stefano Montanelli. 2024. Lexical semantic change through large language models: a survey. *ACM Comput. Surv.*, 56(11).

Francesco Periti and Nina Tahmasebi. 2024. A Systematic Comparison of Contextualized Word Embeddings for Lexical Semantic Change. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4262–4282, Mexico City, Mexico. Association for Computational Linguistics.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: The Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations. 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 1267–1273. Association for Computational Linguistics.

Marko Pranjić, Kaja Dobrovoljc, Senja Pollak, and Matej Martinc. 2024. Semantic change detection for slovene language: a novel dataset and an approach based on optimal transport.

William M. Rand. 1971. Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association*, 66(336):846–850.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Dominik Schlechtweg. 2023. *Human and Computational Measurement of Lexical Semantic Change*. doctoralThesis, University of Stuttgart.

Dominik Schlechtweg, Pierluigi Cassotti, Bill Noble, David Alfter, Sabine Schulte im Walde, and Nina Tahmasebi. 2024. More DWUGs: Extending and evaluating word usage graph datasets in multiple languages.

Dominik Schlechtweg, Enrique Castaneda, Jonas Kuhn, and Sabine Schulte im Walde. 2021a. Modeling Sense Structure in Word Usage Graphs with the Weighted Stochastic Block Model. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 241–251. 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 Fourteenth Workshop on Semantic Evaluation*, pages 1–23, Barcelona (online). International Committee for Computational Linguistics.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2022. DWUG DE: Diachronic Word Usage Graphs for German.

Dominik Schlechtweg, Nina Tahmasebi, Simon Hengchen, Haim Dubossarsky, and Barbara McGillivray. 2021b. DWUG: A large Resource of Diachronic Word Usage Graphs in Four Languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7079–7091, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nina Tahmasebi, Lars Borin, and Adam Jatowt. 2021. *Survey of Computational Approaches to Lexical Semantic Change Detection*, pages 1–91. Language Science Press, Berlin.

Xuri Tang. 2018. A State-of-the-art of Semantic Change Computation. *Natural Language Engineering*, 24(5):649–676.

Frank D. Zamora-Reina, Felipe Bravo-Marquez, and Dominik Schlechtweg. 2022. LSCDiscovery: A Shared Task on Semantic Change Discovery and Detection in Spanish. In *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*, pages 149–164, Dublin, Ireland. Association for Computational Linguistics.

Muhan Zhang and Yixin Chen. 2018. Link prediction based on graph neural networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 5171–5181, Red Hook, NY, USA. Curran Associates Inc.

A Edge induction by language

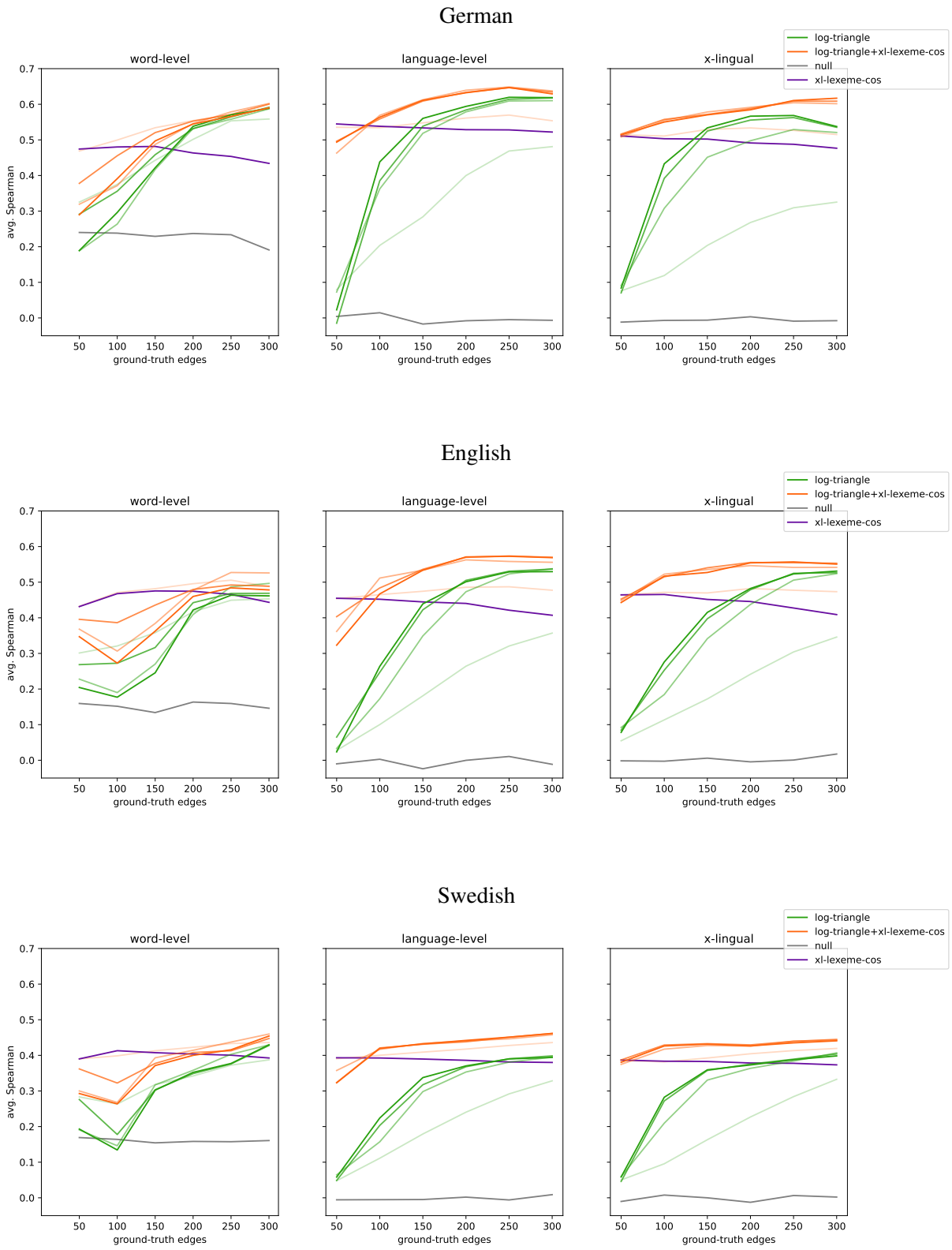


Figure 6: Weighted average Spearman correlations between model predictions and human annotations (see Section 3.3). Here, the scores are computed by considering all annotations from each respective language, and then averaged over 5 folds. For models using `log-triangle` features, inference iterations are shown with increasingly saturated lines, with iteration 4 being the most saturated.

B Experiment 3 data

lemma	usages	edges	% edges
Pachtzins	139	179	1.85
Ackergerät	153	193	1.65
Festspiel	150	193	1.72
aufrechterhalten	140	201	2.05
Ausnahmegesetz	159	202	1.60
weitgreifend	164	205	1.52
Einreichung	160	207	1.62
Unentschlossenheit	176	223	1.44
Mut	200	237	1.19
Frechheit	200	258	1.29
Kubikmeter	200	258	1.29
Truppenteil	200	258	1.29
Entscheidung	200	261	1.31
Gesichtsausdruck	200	265	1.33
Tier	200	272	1.36
Mulatte	200	276	1.38
vergönnen	200	278	1.39
Naturschönheit	198	283	1.44
Lyzeum	200	284	1.42
Behandlung	200	315	1.58
vorliegen	200	331	1.66
Tragfähigkeit	182	337	2.03
voranstellen	200	379	1.90
vorweisen	168	384	2.72
beimischen	200	594	2.97
verbauen	168	1053	7.46

Table 1: Statistics of ground truth data used for training the edge induction models used in Section 6.3.

lemma	usages	edges	% edges
Seminar	50	22	1.76
Spielball	50	26	2.08
Sensation	50	27	2.16
Engpaß	50	32	2.56
Eintagsfliege	50	42	3.36
Manschette	50	43	3.44
Armenhaus	50	44	3.52
artikulieren	50	49	3.92
Knotenpunkt	50	50	4.00
abbauen	50	50	4.00
packen	50	54	4.32
Rezeption	50	54	4.32
Mißklang	50	56	4.48
Abgesang	50	57	4.56
zersetzen	50	60	4.80
überspannen	50	68	5.44
Fuß	50	68	5.44
Titel	50	68	5.44
abgebrüht	50	76	6.08
Schmiere	50	76	6.08
Dynamik	50	81	6.48
abdecken	50	84	6.72
Ohrwurm	50	86	6.88
ausspannen	50	151	12.08

Table 2: Statistics of ground truth data used for edge induction (inference) and clustering in Section 6.3.

C Cluster characteristics

Correlation Clustering

lemma	usage-sense $H(C)$	usage-sense $\mid C \mid$	judgments only $H(C)$	judgments only $\mid C \mid$	judgments only ARI	edge induction $H(C)$	edge induction $\mid C \mid$	edge induction ARI
Spielball	0.17	2	1.17	5	0.05	0.69	4	-0.01
Rezeption	0.37	3	1.80	8	0.08	0.64	4	0.26
Sensation	0.48	3	1.67	7	0.20	1.43	6	0.23
Mißklang	0.51	3	1.46	6	-0.00	-0.00	2	-0.02
artikulieren	0.54	3	1.38	6	0.08	1.36	5	0.17
Abgesang	0.62	3	1.90	9	0.07	1.12	7	0.12
Dynamik	0.64	2	1.76	8	0.13	0.91	4	0.47
Manschette	0.64	4	0.51	4	0.06	0.76	5	0.21
zersetzen	0.68	2	0.68	3	0.34	0.67	3	0.60
Armenhaus	0.68	2	1.76	8	0.11	1.10	5	0.31
Knotenpunkt	0.68	3	1.52	8	-0.01	1.35	6	0.23
Engpaß	0.69	3	1.05	5	0.32	0.88	4	0.34
Ohrwurm	0.69	3	0.68	4	0.66	0.55	3	0.59
Eintagsfliege	0.69	3	0.94	4	0.31	0.68	3	0.50
abgebrüht	0.69	3	1.33	6	0.30	0.92	4	0.78
Titel	0.80	4	0.82	5	0.00	1.03	5	0.07
Seminar	0.92	4	1.29	5	0.12	0.88	4	0.12
packen	1.01	4	1.88	8	0.20	1.45	6	0.41
abbauen	1.04	4	0.92	4	0.32	1.17	5	0.36
ausspannen	1.18	5	1.32	6	0.47	1.23	5	0.63
überspannen	1.22	5	1.38	6	0.20	0.67	3	0.48
abdecken	1.27	6	1.43	6	0.29	0.77	4	0.48
Fuß	1.34	8	1.31	6	0.09	1.43	6	0.48
Schmiere	1.45	8	1.67	7	0.11	0.69	3	0.45

Table 3: Distributional characteristics of `correlation` clusters from Section 6.3 compared to the human *usage-sense* annotation. The *edge induction* column shows the best-performing edge induction model in terms of median ARI (`log-triangle+xl-lexeme-cos`) while *judgments only* is the result of clustering only on ground-truth usage-usage edges. $H(C)$=entropy of the sense/cluster distribution, $\mid C \mid$ = number of senses/clusters, *ARI*=ARI with usage-sense annotation.

SBM-binomial

lemma	usage-sense $H(C)$	usage-sense $\mid C \mid$	judgments only $H(C)$	judgments only $\mid C \mid$	judgments only ARI	edge induction $H(C)$	edge induction $\mid C \mid$	edge induction ARI
Spielball	0.17	2	0.37	2	0.17	0.97	3	0.08
Rezeption	0.37	3	0.85	3	0.21	1.07	4	0.26
Sensation	0.48	3	0.37	2	0.26	1.17	4	0.38
Mißklang	0.51	3	0.53	2	0.21	0.50	2	-0.02
artikulieren	0.54	3	0.23	2	0.14	1.59	5	0.14
Abgesang	0.62	3	0.67	2	0.02	1.37	4	0.08
Dynamik	0.64	2	0.47	2	-0.04	1.16	4	0.49
Manschette	0.64	4	0.67	3	0.38	1.09	4	0.30
zersetzen	0.68	2	0.37	2	0.00	1.08	4	0.60
Armenhaus	0.68	2	0.53	3	0.19	1.16	4	0.33
Knotenpunkt	0.68	3	0.33	2	0.02	1.36	4	0.22
Engpaß	0.69	3	0.17	2	-0.00	1.04	3	0.37
Ohrwurm	0.69	3	0.69	2	0.84	1.09	4	0.63
Eintagsfliege	0.69	3	0.33	2	0.02	1.27	4	0.47
abgebrüht	0.69	3	0.40	2	-0.01	1.37	5	0.60
Titel	0.80	4	0.50	2	-0.02	0.95	3	0.13
Seminar	0.92	4	-0.00	1	0.00	1.11	5	0.11
packen	1.01	4	0.40	2	0.03	1.31	4	0.44
abbauen	1.04	4	0.65	2	0.10	1.33	4	0.38
ausspannen	1.18	5	0.37	2	0.06	1.42	5	0.63
überspannen	1.22	5	0.68	3	0.13	1.29	4	0.43
abdecken	1.27	6	0.81	3	0.23	1.23	4	0.45
Fuß	1.34	8	0.70	3	0.08	1.37	4	0.35
Schmiere	1.45	8	1.21	5	0.29	1.45	5	0.52

Table 4: Distributional characteristics of `sbm-binomial` clusters from Section 6.3 compared to the human usage-sense annotation. The *edge induction* column shows the best-performing edge induction model in terms of median ARI (`log-triangle+xl-lexeme-cos`) while *judgments only* is the result of clustering only on ground-truth usage-usage edges. $H(C)$=entropy of the sense/cluster distribution, $\mid C \mid$ = number of senses/clusters, *ARI*=ARI with usage-sense annotation.

SBM-layers

lemma	usage-sense $H(C)$	usage-sense $\mid C \mid$	judgments only $H(C)$	judgments only $\mid C \mid$	judgments only ARI	edge induction $H(C)$	edge induction $\mid C \mid$	edge induction ARI
Spielball	0.17	2	-0.00	1	0.00	0.44	2	0.30
Rezeption	0.37	3	-0.00	1	0.00	0.49	3	0.70
Sensation	0.48	3	-0.00	1	0.00	0.82	3	0.45
Mißklang	0.51	3	-0.00	1	0.00	1.12	4	0.03
artikulieren	0.54	3	-0.00	1	0.00	1.33	4	0.05
Abgesang	0.62	3	-0.00	1	0.00	1.16	4	-0.02
Dynamik	0.64	2	-0.00	1	0.00	0.93	3	0.48
Manschette	0.64	4	-0.00	1	0.00	0.59	3	0.61
zersetzen	0.68	2	0.28	2	0.03	0.87	4	0.67
Armenhaus	0.68	2	-0.00	1	0.00	0.99	4	0.42
Knotenpunkt	0.68	3	-0.00	1	0.00	1.23	4	0.20
Engpaß	0.69	3	0.17	2	-0.00	0.55	2	0.16
Ohrwurm	0.69	3	0.40	2	0.08	0.53	3	0.15
Eintagsfliege	0.69	3	-0.00	1	0.00	0.89	4	0.35
abgebrüht	0.69	3	0.40	2	-0.01	1.18	4	0.57
Titel	0.80	4	-0.00	1	0.00	1.56	5	0.23
Seminar	0.92	4	-0.00	1	0.00	0.50	2	-0.01
packen	1.01	4	0.37	2	0.04	1.05	3	0.25
abbauen	1.04	4	-0.00	1	0.00	1.05	4	0.21
ausspannen	1.18	5	0.37	2	0.06	1.41	5	0.61
überspannen	1.22	5	0.10	2	0.02	0.95	3	0.38
abdecken	1.27	6	0.55	2	0.33	0.94	4	0.48
Fuß	1.34	8	-0.00	1	0.00	1.02	3	0.29
Schmiere	1.45	8	0.33	2	0.01	1.25	5	0.39

Table 5: Distributional characteristics of `sbm-levels` clusters from Section 6.3 compared to the human usage-sense annotation. The *edge induction* column shows the best-performing edge induction model in terms of median ARI (`log-triangle+xl-lexeme-cos`) while *judgments only* is the result of clustering only on ground-truth usage-usage edges. $H(C)$=entropy of the sense/cluster distribution, $\mid C \mid$ = number of senses/clusters, *ARI*=ARI with usage-sense annotation.