Labeling Results of Topic Models: Word Sense Disambiguation as Key Method for Automatic Topic Labeling with GermaNet

Jennifer Ecker
Leibniz Institute for the German Language
Mannheim, Germany
ecker@ids-mannheim.de

Abstract

The combination of topic modeling and automatic topic labeling sheds light on understanding large corpora of text. It can be used to add semantic information for existing metadata. In addition, one can use the documents and the corresponding topic labels for topic classification. While there are existing algorithms for topic modeling readily accessible for processing texts, there is a need to postprocess the result to make the topics more interpretable and self-explanatory. The topic words from the topic model are ranked and the first/top word could easily be considered as a label. However, it is imperative to use automatic topic labeling, because the highest scored word is not the word that sums up the topic in the best way. Using the lexical-semantic word net GermaNet, the first step is to disambiguate words that are represented in GermaNet with more than one sense. We show how to find the correct sense in the context of a topic with the method of word sense disambiguation. To enhance accuracy, we present a similarity measure based on vectors of topic words that considers semantic relations of the senses demonstrating superior performance of the investigated cases compared to existing methods.

Keywords: automatic topic labeling, word sense disambiguation, topic model, GermaNet

1. Introduction

Topic modeling is a widely used method in natural language processing (NLP) and it is utilized in a variety of applications such as data exploration, enhancing information retrieval or content recommendation for users. Moreover, topic modeling can be used as a method to generate training data from a large corpus for topic classification tasks. Zhou et al. (2023), for example, use topic labels derived from Latent Dirichlet Allocation (LDA) and Gibbs sampling as a part of the training of the TopicBERT-ATP model.

In general, topic modeling is a soft clustering algorithm, where a document can be described by more than one topic (Rüdiger et al., 2022). In some cases, the documents are only described by one topic (hard clustering). As the field of NLP is rapidly evolving, new techniques for topic modeling are also frequently developed. Earlier approaches, for instance Latent Semantic Analysis (LSA) (Deerwester et al., 1990) and especially LDA (Blei et al., 2003), are used in many publications in the digital humanities that deal with topic modeling (Du, 2019), even though it is hard to find the right parameters and suitable number of topics. This leads to incoherent topics and topics that are difficult to understand without preprocessing or postprocessing. Newer approaches like Top2Vec (Angelov, 2020) and BERTopic (Grootendorst, 2022) have the advantages that there's no need to define a specific number of topics in advance, they can be used without lemmatization and stop words, and there is a possibility to decrease the number of topics afterwards.

But what to do with the result of a topic model? The topics are unnamed and consist of a number of associated topic words for each topic. This is where topic labeling comes into play. Topic labeling is the process of assigning descriptive labels to the topics that emerge from a topic modeling algorithm. These topics are often represented as sets of words, which may not be immediately interpretable to humans. Topic labeling aims to provide a succinct and meaningful label for each topic, making it easier to understand the content and context of the topics identified by the model. On the one hand, manually labeling the topics is an option. But that is too time consuming, if there are hundreds of detected topics. The easiest solution is to provide a short list with the highest ranked words for each topic. The reader can then interpret the result on their own. On the other hand, one wants to find a way to do the labeling automatically. One option is to apply a ranking mechanism and pick one of the top n words, usually the first. An alternative is to utilize graph based labeling using DBpedia[1] (Hulpus et al., 2013). There are also approaches that do not directly aim at labeling topics, but improving the result of a topic model. Musat et al. (2011) use WordNet (Miller, 1995) to identify outliers from the top n words as a postprocessing step after applying the topic model. Instead of searching for outliers, it is also possible to find the best topic describing word with a word net. If we combine the two last approaches, we can use the German word net GermaNet (Hamp and Feldweg, 1997; Henrich and Hinrichs, 2010) to do the

[1] https://www.dbpedia.org/

LREC-COLING 2024, pages 10014–10022

graph-based labeling instead of using DBpedia.
Finding the words of a topic in GermaNet is partly challenging, because there are polysemous words, i.e., they have two or more meanings, which generally are differently represented in word nets. Word sense disambiguation is therefore employed to find the correct sense in the context of a topic. In this paper, we combine word sense disambiguation with topic modeling. Additionally, we present a method that works without the use of glosses from the word net, as they are not always available for each word. Instead, it is based on topic words and their semantic relations. Therefore, we analyze the topic words of one topic with two different measures, one based on paths in GermaNet and word frequencies, which according to previous work leads to the best results (see Section 2.2.1), and the other on word embeddings, the more recent measure (see Section 3.3.1). For the latter, we compare mean vectors of sets of the different senses of a word and their hypernyms and hyponyms found in GermaNet.
This paper is structured as follows. We first summarize different methods of topic modeling and then outline the literature that assesses their comparative effectiveness (Section 2.1). Second, we report the steps of selected approaches that help labeling the result of a topic model, i.e., word sense disambiguation in knowledge bases (Section 2.2). Next, our approach of automatic topic labeling is presented, starting with the utilized data set for the topic model (Section 3.1), the topic model itself (Section 3.2), the first part of the procedure to label the topics of the topic model (Section 3.3), and an evaluation of the used measures (Section 3.4). We conclude with a discussion of the used methods for word sense disambiguation (Section 3.5), sum up the next steps left for future work (Section 4) and recap the contributions of this paper (Section 5).

2. Related Work

In this section, we describe methods and resources that are indispensable for the task of automatic topic labeling. First of all, we summarize the most important findings in the field of topic modeling to justify the choice of the used topic model in our research.

2.1. Modeling Topics

Finding the best topic model is getting more complex regarding the fact that there are many models to use off the shelf. Rüdiger et al. (2022) review four of the earlier approaches, two non-probabilistic (LSA and NMF, Non-negative Matrix Factorization, (Lee and Seung, 2000)) and two probabilistic (PLSA, Probabilistic Latent Semantic Analysis, (Hofmann, 1999) and LDA), applied on English Wikipedia articles. For a small number of topics (less than 20) NMF works best. For more topics (more than 20) LDA outperforms the other approaches. Overall, by increasing the number of topics they observe a slow loss of predictive power of all algorithms. Additionally, they also show a correlation between increasing the number of topics and an increasing difficulty to tell the topics apart. Another problem is the high sensitivity to cluster sizes, text lengths, text characteristics, and text preprocessing.
Egger and Yu (2022) compare LDA and NMF with Top2Vec and BERTopic, two of the latest topic models, on a corpus of twitter posts. Both Top2Vec and BERTopic work with embeddings, but differ in the point that BERTopic uses the class-based TF-IDF to extract the topic words. Egger and Yu (2022) conclude that the topics from Top2Vec are more overlapping and a topic may contain not only one concept making BERTopic the better choice for their data. Between LDA and NMF they recommend NMF and between LDA and Top2Vec they advocate the latter on grounds of better results. To sum up, it is advisable to reflect the pros and cons of the different algorithms to choose the best model for the underlying data. To improve the results of Top2Vec and BERTopic, there is an option to reduce the topics afterwards, yet Egger and Yu (2022) state that this is subject to human interpretation. The Top2Vec algorithm[2] utilized in the research we carried out in this paper works best on very large data sets (more than 1 000 documents) and is applicable to multiple languages within a corpus (Egger and Yu, 2022).

2.2. Labeling Topics

Labeling the topics is the next step after applying the topic modeling algorithm. As described in Section 1, we can use a word net such as GermaNet to automatically label the topics. GermaNet is a lexical-semantic net for German, in which words are grouped based on their semantic relatedness. It shares the basic framework with the Princeton WordNet, but is an independent net that is created from scratch. The nodes in a word net are concepts of words, so-called synsets with the same meaning (synonyms). Polysemous words are represented as different senses of a word, e.g., the word *Album* in German refers to a music album or a book with collected things like pictures or stamps (collector's album). Additionally, the senses in GermaNet include mappings to Wiktionary descriptions (glosses) (Henrich, 2015). The synsets are interlinked by lexical semantic relations (e.g., hypernymy or hyponymy). Hulpus et al. (2013) use a knowledge graph (DBpedia) in-

[2] https://top2vec.readthedocs.io/en/latest/Top2Vec.html#the-algorithm

stead of a word net to do the topic labeling. Their steps include word sense disambiguation, graph extraction, and graph-based labeling.

2.2.1. Applying Word Sense Disambiguation

Word sense disambiguation is a method to determine which sense of a word in context is the correct one (Navigli, 2009). Methods for word sense disambiguation include supervised, unsupervised, and knowledge-based approaches. Navigli (2009) points out that supervision and knowledge can be used to varying extents to maximize the performance. Furthermore, Bevilacqua et al. (2021) present a survey of approaches for word sense disambiguation and point out current limitations of the task.

In the following, we first focus on knowledge-based disambiguation. There are several similarity measures to calculate the relatedness of concepts and to disambiguate polysemous words. McInnes and Pedersen (2013) classify semantic similarity measures into path-based measures, corpus-based information content (IC) measures, and taxonomy-based IC measures. Path-based measures compute the relatedness based on the hierarchy between two concepts, specifically on the shortest path (Leacock and Chodorow, 1998; Wu and Palmer, 1994). Corpus-based IC measures extend path information with word frequencies from a corpus (Resnik, 1995; Jiang and Conrath, 1997; Lin, 1998). Taxonomy-based IC measures compute the concept's information content by means of the location in the hierarchy (Sánchez et al., 2011). There also are relatedness measures that calculate relatedness of two concepts based on the overlap of their descriptions (Lesk, 1986). In their evaluation of the above mentioned measures with biomedical texts, McInnes and Pedersen (2013) found a higher disambiguation accuracy for corpus-based and taxonomy-based IC measures by comparison of all measures. Batista et al. (2012) compare IC-based measures on news articles and conclude that the Jiang-Conrath measure shows the best result. The Jiang and Conrath distance (Jiang and Conrath, 1997) of two nodes is computed by the sum of the information content of two concepts minus the double information content of their least common subsumer (see Equation 1). Two concepts are more related the smaller this distance is (Henrich, 2015).

$$Dist(w_1, w_2) = IC(c_1) + IC(c_2) - 2xIC(LSuper(c_1, c_2)) \quad (1)$$

Unsupervised disambiguation methods typically assume that the same sense of a word has similar neighboring words (Navigli, 2009). Those fully unsupervised methods (knowledge-lean approaches) do not make use of dictionaries or ontologies (Pedersen, 2006). To cluster words based on similar contexts, corpora are used instead. The different meanings of a word are discriminated with the help of these clusters. An example for this approach is Schütze (1998). In contrast, supervised disambiguation mainly uses machine learning techniques to classify the correct sense of a word using a classifier trained with sense-annotated data (Navigli, 2009). A recent example of a neural architecture is EWISER (Bevilacqua and Navigli, 2020) that incorporates WordNet graph data directly into its neural structure, enabling it to leverage relational information with pretrained synset embeddings.

3. Automatic Topic Labeling

In this section, we present our approach of automatically labeling topics. First, we describe the preliminary work including the data set and the algorithm for topic modeling. Second, we present our work of finding the best label for the generated topics and their corresponding topic words.

3.1. Data Set

The used data are articles from the newspaper corpus Mannheimer Morgen volume 20 (M20) as a part of DeReKo (The Mannheim German Reference Corpus[3], Kupietz et al. (2020)). The corpus is available in a specific format (the IDS text model[4] - I5 format) from the Leibniz Institute for the German Language. First, the various articles need to be extracted from the large XML file, which is hierarchically structured as follows:

- corpus level,
- document level,
- text level.

In the case of M20, the document level represents the month, in which an article is published. We extract the text from the text level and save it in a separate text file for each article (44 383 in total).

3.2. Topic Model Algorithm

As the algorithm for topic modeling we choose Top2Vec over BERTopic, because BERTopic classifies some documents as outliers and our goal is to have a specific topic label for every document. Additionally, Top2Vec is well suited for processing a big amount of documents and it computes reliable results. On another note, topic reduction can be performed to merge close topics.

[3] https://www.ids-mannheim.de/en/digspra/corpus-linguistics/projects/corpus-development/

[4] https://www.ids-mannheim.de/en/digspra/corpus-linguistics/projects/corpus-development/ids-text-model/

We apply the Top2Vec python library[5] on the data set and make modifications in the code of the library to use it for the German language[6]. There are different models in the library to use. We choose the embedding model *doc2vec* to apply on the data set. The speed parameter is set to *deep-learn* and for all other parameters the default is used. For 44 383 documents the topic model identifies 348 topics. Figure 1 shows the word cloud for topic 0 with the topic describing words in lowercase.

The cosine similarity that describes the semantic similarity between the document vector and the topic vector is assigned to the documents. It is possible to specify the number of topics per document that are returned. We only select the topic with the highest similarity score and thus employ hard clustering.

3.3. Method of Topic Labeling

After manual inspection of the top 20 words of a topic, not every highest ranked word describes the topic ideally. Instead of using the highest ranked topic word, we label the topics with GermaNet. The identification of the label with GermaNet includes the following steps:

- word sense disambiguation,
- graph extraction,
- graph-based labeling.

In this paper, we only focus on word sense disambiguation applied on topic models.

3.3.1. Word Sense Disambiguation

In the domain of word sense disambiguation, achieving the accurate sense of a word typically involves the contextual analysis of the word alongside the glosses representing potential word senses (i.e., the synsets found in GermaNet). Again, we cannot apply a gloss-based approach, as there is no gloss available for every synset in GermaNet.

A topic modeling algorithm assigns words to topics based on their co-occurrence patterns within the text corpus. This means that words frequently appearing together are grouped into the same topic. Consequently, after applying the algorithm, we obtain sets of words associated with each topic. Although these sets do not provide the exact sentence-level context in which the words appear, they do offer a contextual framework. We leverage these topic-associated words as context for word sense disambiguation, because they represent co-occurring terms that share semantic relationships within a specific thematic context. By employing these topic words, we aim to disambiguate words that possess multiple synsets in GermaNet, utilizing the contextual cues provided by their topic associations.

The first step is to find the topic words in GermaNet (using the release 18.0 of May 2023). The second step is to identify which words have only one synset and which words have more than one synset. There certainly are words that do not appear in GermaNet at all, like names (e.g., music band names) and not all abbreviations can be found (e.g., *MHC* for *Mannheimer Hockeyclub* 'hockey club of the city Mannheim'). The third step is to disambiguate the words that have many synsets with the help of the words that have only one synset. The logic behind this is that each topic consists of words that are semantically related to each other and thus contribute to disambiguation. With their help, we find the correct word sense and we extract the GermaNet ID of each topic word. To sum up, we do not use the usual approach of word sense disambiguation, but a modified interpretation of the context of a word to get the correct word sense. Specifically, instead of using the sentence in which a word occurs, we use the other words it appears with in a topic.

For the disambiguation process, we use the top 20 words of a topic. For topic 0 there are eight words (*Schlagzeuger* 'drummer', *Pop* 'pop', *Musiker* 'musician', *Gitarrist* 'guitarist', *Soul* 'soul music', *Gitarre* 'guitar', *Jazz* 'jazz', and *Schlagzeug* 'drums') that are non-ambiguous and six that are ambiguous (*Album* 'album', *Sound* 'sound', *Song* 'song', *Rock* 'rock', *Blues* 'blues', and *Band* 'band'). The remaining six words out of 20 topic words are German plurals or English words not found in GermaNet. We apply word sense disambiguation to adjectives, adverbs, nouns, and verbs, because they are represented in GermaNet.

As the first similarity measure to find the correct sense of a word, we use the Jiang and Conrath similarity (JC-similarity). Therefore, the distance measure of Equation 1 is subtracted from the maximum possible distance of any two concepts[7] (Henrich, 2015)[8]. For this paper, we calculate the similarity of the first synset with the first non-ambiguous topic word, then with the second ambiguous topic word, and so on. After that, the average of all values for the first synset is calculated. The same procedure is followed with the next synset in order to later compare which average similarity is the highest to choose one synset.

[5] https://top2vec.readthedocs.io/en/latest/Top2Vec.html

[6] Setting *deacc=True* to *deacc=False* inside the default tokenizer function of the Top2Vec python file lets us work with German umlauts.

[7] https://pypi.org/project/germanetpy/

[8] For more information about it see the dissertation.

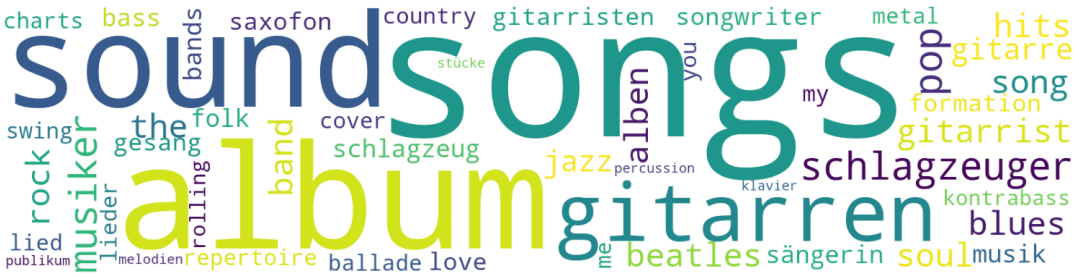

Figure 1: Word cloud of topic 0 from the top 50 words.

Table 1 shows the mean similarity of all non-ambiguous topic words compared to each meaning of a polysemous word. The best score and the correct sense of a word are marked in bold. To illustrate the meaning of each synset, we added its hypernym to the table. In the context of topic 0, the correct synset for the word *Album* is *s70622* and the correct synset for the word *Rock* is *s27229*. As we can see from the table, the wrong synset for the word *Album* has the highest similarity score, whereas for the word *Rock* the correct synset has the highest score. For the word *Album* the two scores are close.

Word	ID	Hypernym	Similarity
Album	s10914	Buch 'book'	**0.628425**
	s70622	Tonträger 'sound carrier'	0.61675625
Rock	s7008	Jacke 'jacket'	0.60974125
	s7160	Oberbekleidung 'outwear'	0.61612125
	s27229	Musikstil 'style of music'	**0.71612375**

Table 1: Comparison of the JC-similarity for the polysemous words *Album* and *Rock* with the non-ambiguous words *Schlagzeuger* 'drummer', *Pop* 'pop', *Musiker* 'musician', *Gitarrist* 'guitarist', *Soul* 'soul music', *Gitarre* 'guitar', *Jazz* 'jazz', and *Schlagzeug* 'drums'.

As the JC-similarity does not reliably pick the correct meaning of a polysemous word and the similarity scores are close, we choose to combine two methods for word sense disambiguation to improve the results: knowledge-based disambiguation and unsupervised disambiguation. It is a combination of searching the words and their hypernyms and hyponyms in GermaNet and word embeddings. For this, we use a German word embedding model[9]. Then, we compare the sum of the vectors of all the topic words that only have one synset with the vector of each sense of a polysemous word with their direct hypernyms and direct hyponyms (see Table 2). The comparison is made with the cosine similarity between the mean vectors (m_1, m_2) of two sets (ws_1, ws_2) (see Equation 2).

$$\text{Sim}(m_1, m_2) = cos(\theta) = \frac{\vec{m_1} \cdot \vec{m_2}}{||\vec{m_1}|| \times ||\vec{m_2}||}$$
$$m_i = \sum_{j}^{N} ws_{i,j} \quad (2)$$

We choose the sense with the highest cosine similarity. Now, the correct sense for the word *Album* is selected and for the word *Rock* the difference between the senses is even more distinct compared to applying only the JC-similarity.

3.4. Evaluation

To evaluate the performance of the two methods, we first manually assign IDs of GermaNet to the polysemous topic words to create a gold standard. The annotation was made by the author of this paper due to various practical considerations. Given the scope and resources available for this research, involving external annotators was not feasible within the given timeframe and budget constraints. While acknowledging the importance of inter-annotator agreement as a measure of reliability, the decision to perform the annotations in this way was a pragmatic one.
Value 0 is assigned, if the topic word does not fit into the topic or there is no suitable sense available in GermaNet. If all of the senses fit into the topic, *all* is assigned (e.g., *Handball* as a kind of sports and the ball used for playing handball). Sometimes, more than one sense is correct, but mostly

[9] https://huggingface.co/Word2vec/german_model

Word	ID	Hypernym	Hyponyms	Similarity
Album	s10914	Buch 'book'	Sammelalbum, Erinnerungsalbum 'collector's album', 'scrapbook'	0.49917594
	s70622	Tonträger 'sound carrier'	Musikalbum, Tributealbum, Debütalbum, Plattenalbum, Doppelalbum 'music album', 'tribute album', 'debut album', 'disc album', 'double album'	**0.5780323**
Rock	s7008	Jacke 'jacket'	Uniformrock, Gehrock, Frack 'tunic', 'frock', 'tailcoat'	0.48681718
	s7160	Oberbekleidung 'outwear'	Tellerrock, Schottenrock, Sommerrock, ... 'circle skirt', 'kilt', 'summer skirt'	0.5122094
	s27229	Musikstil 'style of music'	Heavy Metal, Hardrock, Metal, ... 'heavy metal', 'hard rock', 'metal'	**0.8165419**

Table 2: Comparison of vectors for the polysemous words *Album* and *Rock* with the vector of the non-ambiguous words *Schlagzeuger* 'drummer', *Pop* 'pop', *Musiker* 'musician', *Gitarrist* 'guitarist', *Soul* 'soul music', *Gitarre* 'guitar', *Jazz* 'jazz', and *Schlagzeug* 'drums'.

there is only one sense selected as the label. After filtering the topic words that are assigned the value 0, there are 1 105 labeled topic words left. The average number of potential senses is 2.77. Then, we compare the outcome of the JC-similarity and the vector-similarity with a baseline. For the baseline, a sense is randomly[10] selected from all found word senses for a specific polysemous topic word in GermaNet. We do not use the more common approach that utilizes the most frequent baseline, because there is no information about it available in GermaNet. Employing a corpus analysis could potentially help identify the most frequent sense. Hence, opting for random synsets was a more straightforward choice for establishing the baseline. We calculate accuracy, precision, recall, and F1-score for baseline, JC-similarity, and vector-similarity[11]. The results[12] are presented in Table 3 with the best score for each metric in bold. For precision, recall, and F1-score we compute the weighted average. Both, JC-similarity and vector-similarity are better than the baseline in all metrics with the vector-similarity having the highest scores. Please note that the scores are dependent on the way the labels are created. As 25 percent of the data is labeled as *all*, indicating that every decision made by a method is considered correct for these cases, the baseline scores are relatively high regarding the average number of potential word senses.

[10]Each time GermaNet is loaded the order of synsets for an ambiguous word varies and we select the first synset.

[11]A part of the code (the evaluation) and a file with the selected sense for each method are available on GitHub: https://github.com/jecker94/WSD_TM.

[12]All results are rounded by the used sklearn metric function (https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html).

Method	Accuracy	Precision	Recall	F1
Baseline	0.58	0.61	0.58	0.59
JC-sim	0.74	0.75	0.74	0.74
Vector-sim	**0.84**	**0.83**	**0.84**	**0.83**

Table 3: Evaluation of the different methods with accuracy, precision, recall, and F1-score.

3.5. Discussion

In this section, we discuss the benefits and drawbacks of the methods used for word sense disambiguation. In general, the JC-similarity is better than path-based measures, because it incorporates word frequencies. Here, we use the word frequency list provided by GermaNet. Nevertheless, it is not powerful enough to reliably predict the correct sense as we can see from the evaluation in Section 3.4. Comparing the JC-similarity to the vector-similarity, the vector-similarity is about 10 percent better in all scores. Moreover, vectors are more powerful the bigger the underlying data set is. Because of this, we use the existing German Word2Vec model and do not use the M20 data to build a model from scratch. The downside of both measures is that they do not distinguish between the different senses of polysemous words. Words in the corpus are counted (for the JC-similarity) and vectors of Word2Vec are computed regardless of the context resulting in a 1:1 relation, even though there are different meanings based on the contexts. For this reason, we combine the single word vector with the vectors of its hypernyms and hyponyms in the second method. Accordingly, we get a mean vector that represents the context of a sense. While creating the gold standard, we realize that even for a human it is not always easy to

select the best word ID from GermaNet. Accordingly, the scores from the approach with vectors can be classified as sufficiently good. To sum up, GermaNet provides a solution to apply word disambiguation on topic words. Its structural framework enriches the disambiguation process by incorporating valuable contextual cues to identify the appropriate synset amidst polysemous words. Despite the absence of glosses for individual synsets, the direct hypernyms and hyponyms of the synsets resolve this limitation, providing more semantic information on the synsets to facilitate accurate synset selection.

4. Future Work

The approach in this paper is a first step to directly label the topics of a topic model. As the topic words are already coherent, there is no need to improve them, but to choose a suitable label. We aim to provide a succinct and meaningful label for each topic, facilitating comprehension of the content and context of the topics discerned by the topic model. The effectiveness of automatic topic labeling, particularly in word sense disambiguation, heavily relies on the chosen topic modeling method. If the topic words are not coherent and contain outliers, the polysemous words of a topic might not be mapped to the sense fitting in the context. The next steps include to see if the results of the topic model can be improved with lemmatization of the topic words. We also considered applying hierarchical topic reduction despite the disadvantage that the cosine similarity between the documents and the topic vector gets smaller, the more topics are merged. Another option is to compare Top2Vec with BERTopic. With BERTopic, we would get an outlier topic with all documents that do not fit into a topic and could label it as miscellaneous. This could lead to an overall higher semantic similarity between each document and topic vector, as BERTopic is classified by Egger and Yu (2022) as the better algorithm for their data. One valuable aspect to identify is, if it creates a similar number of topics or if it achieves to sum up the topics in a more efficient way.

Another approach is to filter documents with a small word count, because there needs to be enough content for the topic model to reliably assign a topic. However, this violates our goal to have a specific label for every document for the task of enriching metadata. If we concentrate on using topic modeling as a first step for the task of topic classification, we do not need a label for each document and therefore we can filter small documents. And thus later assign a topic based on the classification for the missing documents.

Beyond word sense disambiguation, the next step of automatically labeling the topics is to explore graph extraction and graph-based labeling. This method is based on the assumption that related concepts of a topic lie close within a graph of a knowledge base (Hulpus et al., 2013).

5. Conclusion

This paper demonstrates the efficiency of utilizing topic words for disambiguating polysemous words. Notably, it underscores that disambiguation can be achieved without relying on complete sentences, particularly in scenarios where contextual sentences (glosses) are absent in a word net. Moreover, the paper highlights the efficacy of leveraging the structural relationships within GermaNet, such as hypernyms and hyponyms, to enhance the disambiguation of polysemous words. We show that using vectors as similarity measure is superior to IC-based calculations, leading to higher accuracy and F1-scores. In this paper, we used simple static word embeddings, but contextualized word embeddings could contribute to an even better performance of the vector-similarity. The overall effectiveness of using GermaNet in this context depends on the next step, that is generating adequate labels for the topics with a graph representation from GermaNet. At the time of the paper submission, we have unfortunately not been able to test the generating of adequate labels, as there are certain considerations to take into account, e.g., how much of the graph from GermaNet should be extracted and which centrality measure should be applied.

6. Acknowledgements

This publication was created in the context of the work of the association German National Research Data Infrastructure (NFDI) e.V. NFDI is financed by the Federal Republic of Germany and the 16 federal states, and the consortium Text+ is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - project number 460033370. The authors would like to thank for the funding and support. Furthermore, thanks also include all institutions and actors who are committed to the association and its goals.

7. Bibliographical References

Dimo Angelov. 2020. Top2vec: Distributed representations of topics. *arXiv preprint arXiv:2008.09470*.

David S Batista, João D Ferreira, Francisco M Couto, and Mário J Silva. 2012. Toponym disambiguation using ontology-based semantic similarity. In *Computational Processing of the Portuguese Language: 10th International*

Conference, PROPOR 2012, Coimbra, Portugal, April 17-20, 2012. Proceedings 10, pages 179–185. Springer.

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864, Online. Association for Computational Linguistics.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. Recent trends in word sense disambiguation: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4330–4338. International Joint Conferences on Artificial Intelligence Organization.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Xinxiong Chen, Zhiyuan Liu, and Maosong Sun. 2014. A unified model for word sense representation and disambiguation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1025–1035.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Keli Du. 2019. A survey on LDA topic modeling in digital humanities. *Book of Abstracts DH2019*, 10:H9UYPI.

Roman Egger and Joanne Yu. 2022. A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Frontiers in sociology*, 7:886498.

Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Birgit Hamp and Helmut Feldweg. 1997. Germanet-a lexical-semantic net for german. In *Automatic information extraction and building of lexical semantic resources for NLP applications*.

Verena Henrich. 2015. *Word sense disambiguation with GermaNet*. Ph.D. thesis, Universität Tübingen.

Verena Henrich and Erhard W Hinrichs. 2010. Gernedit-the GermaNet editing tool. In *ACL (System Demonstrations)*, pages 19–24.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57.

Ioana Hulpus, Conor Hayes, Marcel Karnstedt, and Derek Greene. 2013. Unsupervised graph-based topic labelling using dbpedia. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 465–474.

Jay J Jiang and David W Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*.

Claudia Leacock and Martin Chodorow. 1998. Combining local context and wordnet sense similarity for word sense identification. wordnet, an electronic lexical database. *The MIT Press*.

Daniel Lee and H Sebastian Seung. 2000. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13.

Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Icml*, volume 98, pages 296–304.

Bridget T McInnes and Ted Pedersen. 2013. Evaluating measures of semantic similarity and relatedness to disambiguate terms in biomedical text. *Journal of biomedical informatics*, 46(6):1116–1124.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Claudiu Musat, Julien Velcin, Marian-Andrei Rizoiu, and Stefan Trausan-Matu. 2011. Concept-based topic model improvement. *Emerging intelligent technologies in industry*, pages 133–142.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, 41(2):1–69.

Ted Pedersen. 2006. Unsupervised corpus-based methods for WSD. *Word sense disambiguation*, pages 133–166.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*.

Matthias Rüdiger, David Antons, Amol M Joshi, and Torsten-Oliver Salge. 2022. Topic modeling revisited: New evidence on algorithm performance and quality metrics. *Plos one*, 17(4).

David Sánchez, Montserrat Batet, and David Isern. 2011. Ontology-based information content computation. *Knowledge-based systems*, 24(2):297–303.

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

Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. *arXiv preprint cmp-lg/9406033*.

Yuxiang Zhou, Lejian Liao, Yang Gao, Rui Wang, and Heyan Huang. 2023. TopicBERT: A topic-enhanced neural language model fine-tuned for sentiment classification. *IEEE Transactions on Neural Networks and Learning Systems*, 34(1):380–393.

8. Language Resource References

Marc Kupietz and others. 2020. *Deutsches Referenzkorpus - DeReKo-2020-I*. Institut für Deutsche Sprache. [link].