Assessing the Significance of Encoded Information in Contextualized Representations to Word Sense Disambiguation

Deniz Ekin Yavas
Heinrich Heine University Düsseldorf
deniz.yavas@hhu.de

Abstract

The similarity of representations is crucial for WSD. However, a lot of information is encoded in the contextualized representations, and it is not clear which sentence context features drive this similarity and whether these features are significant to WSD. In this study, we address these questions. First, we identify the sentence context features that are responsible for the similarity of the contextualized representations of different occurrences of words. For this purpose, we conduct an explainability experiment and identify the sentence context features that lead to the formation of the clusters in word sense clustering with CWEs. Then, we provide a qualitative evaluation for assessing the significance of these features to WSD. Our results show that features that lack significance to WSD determine the similarity of the representations even when different senses of a word occur in highly diverse contexts and sentence context provides clear clues for different senses.

1 Introduction

Contextualization is a powerful tool as it enables us to capture sentence context. This is crucial especially in word sense disambiguation (WSD) because sentence context provides valuable information for resolving lexical ambiguity in both NLP and human language processing.

The similarity of representations is crucial for WSD. With contextualization, we expect the representations of different occurrences of the same sense to be similar to each other. This is based on the assumption that different senses of a word occur in different contexts and sentence context contains explicit clues that signal one of the senses of the word. Consider the sentences in (1) that demonstrate two senses of 'bank'. In both sentences, some words successfully signal each sense of the word; in (1-a), the words 'money' and 'withdraw' and in (1-b), the words 'picnic' and 'river'.

(1) 'bank' (homonymy):
   a. *financial institution:*
      I went to the **bank** to withdraw money.
   b. *geographical feature:*
      They had a picnic by the river **bank**.

(2) 'pass' (polysemy):
   a. *go across or through:*
      She **passed** through towns.
   b. *move past:*
      She **passed** the bakery on her way.

However, in practice, we lack clarity on which specific sentence context features are responsible for the similarity of the contextualized representations. It has been shown that a wide variety of information is encoded in the contextualized representations (Sajjad et al., 2022) and using contextualized word embeddings (CWEs) of pre-trained language models alone does not achieve good performance in unsupervised settings (Yenicelik et al., 2020).

The purpose of this study is to investigate which sentence context features determine the similarity of the representations of different occurrences of words and whether these features are significant to WSD. By doing so, we aim to provide a clearer understanding of contextualized representations in terms of their ability to capture different meanings of words. For this purpose, we conduct an explainability experiment. We focus on word sense clustering with CWEs of BERT (Devlin et al., 2019) and identify sentence context features that lead to the formation of the clusters. This way, we determine which features drive the similarity of the representations.

Our cluster explainability method follows several steps and is depicted in Figure 1. We start by performing word sense clustering with CWEs and cluster the sentences of a word. Our aim is essentially to reverse the word sense clustering process

Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language, pages 42–53

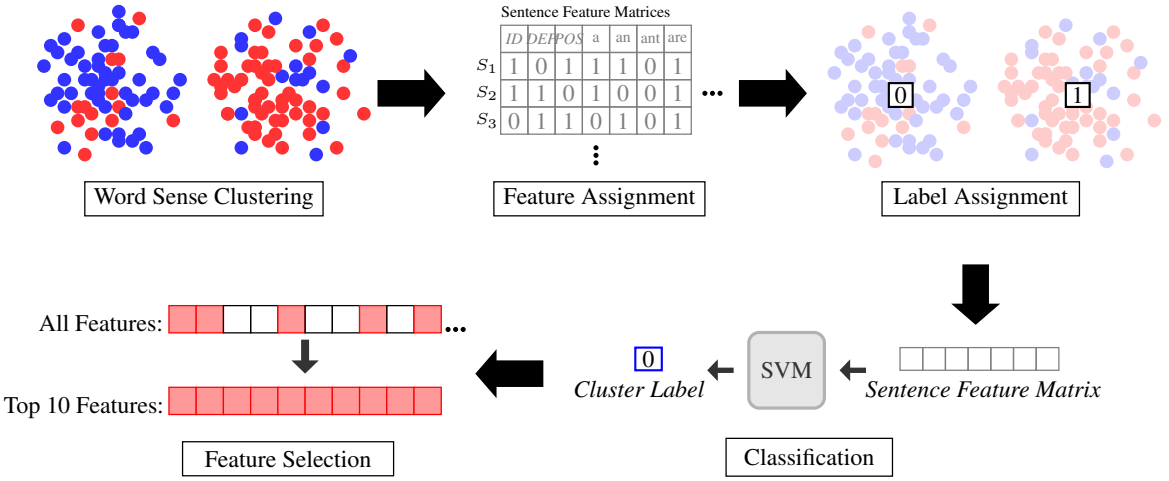

Figure 1: Cluster Explainability Method: i) Perform Word Sense Clustering with CWEs, ii) Represent each sentence with sentence context features, iii) Assign each sentence a label based on the cluster it belongs to, iv) Train a classifier to predict the clusters that sentences belong to based on their sentence context features, v) Apply feature selection to determine the sentence context features responsible for the performance of the classifier.

and recreate the clusters with a classification task. For this purpose, we represent each sentence with its sentence context features and assign it a label based on the cluster that it belongs to. Then, we train a classifier to predict the cluster labels of the sentences based on their sentence context features. As the final step, we apply a feature selection algorithm to the classifier to determine which features are the most relevant for the performance of the classifier. This tells us which sentence context features lead to the formation of the clusters. We use this method to identify the features that lead to the clusters for each word in our dataset. Finally, we assess the significance of these features to WSD for each word through qualitative evaluation.

In this study, we distinguish two types of lexical ambiguity; homonymy and polysemy. The most distinctive feature that distinguishes these two types is the semantic relatedness of their senses (Klepousniotou, 2002; Klepousniotou and Baum, 2007; Klepousniotou et al., 2008, 2012). Homonyms have less semantically related senses compared to polysemes. As a result, homonyms occur in more diverse contexts. Consider the examples in (1) and (2). The provided senses of 'bank' are homonymous, whereas those of 'pass' are polysemous. The senses of 'bank' are not related semantically and the noun co-occurs with semantically different words in its different senses ('money', 'withdraw' vs. 'river', 'picnic'). However, this is not true for 'pass'. The meaning difference between the senses of 'pass' is less clear and the verb co-occurs with words that are similar in meaning in its different senses, specifically words that are related to a location.

Their inherent differences also result in differences in NLP performance. For example, WSD performance is better with homonyms (Nair et al., 2020; Haber and Poesio, 2021) and contextualization affects homonyms more (Sevastjanova et al., 2021) compared to polysemes. Therefore, it's important to consider that not all words present an equal challenge for WSD. In addition to that, it's important to consider that different lexical ambiguity types have different relations to context, and as a result, the information that is required for their disambiguation might not always be the same. In this study, we expect different results for each type. Considering that homonyms occur in more diverse contexts, we expect sentence context to provide clearer clues for their different senses and the similarity of the representations to be affected by these clues.

Our results show that the sentence context features that are responsible for the similarity of the representations and lead to the formation of clusters lack significance to WSD in most cases. This is true for both lexical ambiguity types. Even with homonyms—where different senses of a word occur in highly diverse contexts and sentence context provides clear clues for different senses—the similarity of the representations does not arise from the significant features.

2 Related Work

Studies have shown that the similarity of the embeddings is primarily influenced by the sentence context rather than the meaning of words. Ethayarajh (2019) have shown that words have different representations based on their contextual variation, rather than their meaning variation. Similarly, Garcia (2021) have shown that the similarity between a word and its synonym is lost when the sentence context is identical for different words; the similarity between a word and a random word is not different from the similarity between a word and its synonym when their sentence contexts are similar.

One reason for this is that a lot of information is encoded in the contextualized representations and they affect the similarity of the representations. Sajjad et al. (2022) have shown that the information encoded in the contextualized representations can be explained to some extent by semantic, morphological, syntactic, and lexical concepts. These concepts include the words' POS tags, CCG super-tags, ngrams, casings, WordNet concepts, and so on. Additionally, Mickus et al. (2020) have shown that the similarity of the representations is affected by the segment embeddings that the model assigns to tokens to indicate their sentences. In this study, this is not an issue because we only use one sentence as an input. However, we aim to investigate whether a similar effect can be found for the positional encoding, resulting words in the same position having similar representations.

Clustering reveals these similarities within the representations. Sajjad et al. (2022) have shown that the clusters of contextualized representations overlap with the concepts that are found to be encoded in the representations. Furthermore, it has been shown that word sense clustering with CWEs of the BERT model doesn't achieve good performance and sentence context similarities have been observed within the clusters (Yenicelik et al., 2020). The effects of the sentence context have been also observed in similarity ranking for WSD with CWEs of BERT (Gessler and Schneider, 2021). However, there hasn't been any effort to systematically explain the relation between sentence context and contextualized representations of different occurrences of the same word and with a focus on WSD. This study aims to fill this gap.

Finally, the studies that distinguish different types of lexical ambiguity have shown that WSD performance with CWEs changes depending on the type and it is easier to disambiguate homonymy than polysemy (Nair et al., 2020; Haber and Poesio, 2021). Similarly, contextualization of BERT affects different types differently and homonyms are affected more by contextualization due to the fact that they occur in more diverse contexts (Sevastjanova et al., 2021). In this study, we also expect the results to be different for each type. In the case of homonymy, we expect sentence context to provide clearer clues for different senses and the similarity of the representations to be affected by these clues.

3 Data

We use SemCor (Miller et al., 1993) which provides sentences that are annotated with WordNet senses for a wide variety of words (Fellbaum, 2010) for English. We restrict our focus to nouns and verbs. A word can be both homonymous and polysemous at the same time because different senses of a word can have different relations, e.g. two senses can be homonymous while another two can be polysemous. Because of this, we don't focus on homonymous or polysemous words but sense groups of words. These sense groups are formed by grouping the senses of a word according to their relations, so we end up with sense groups in which all pairs are homonymous or polysemous to each other.

In order to decide if a sense pair is homonymous or polysemous, we use the data provided in Nair et al. (2020). This data contains semantic relatedness judgment scores for a subpart of WordNet. Semantic relatedness determines where on the homonymy-polysemy continuum a word is (Klepousniotou, 2002; Klepousniotou and Baum, 2007; Klepousniotou et al., 2008, 2012) and semantic relatedness judgments of speakers overlap with different types of lexical ambiguity (Klepousniotou et al., 2008; Nair et al., 2020). In Nair et al. (2020), semantic relatedness judgement scores are collected for each sense pair from several speakers. We use the average of the scores for each sense pair to decide if the word is homonymous or polysemous in those senses. We consider the sense pairs that have a distance over 0.8 as homonymy pairs and a distance below 0.5 as polysemy pairs.[1]

[1]Psycholinguistics studies have shown that some polysemy types show homonymy-like behaviors and have less semantic relatedness (Klepousniotou and Baum, 2007; Klepousniotou et al., 2008). Due to this, we leave a certain range out to avoid these mixed types.

Lexical Ambiguity	Clustering		Sentence-Feature Classifiers										
	Sense Group #	ARI	Sense Group #	*WSD* F1	*WSD* P	*WSD* R	*Cluster* F1	*Cluster* P	*Cluster* R	*10-Cluster* F1	*10-Cluster* P	*10-Cluster* R	*10-Rand.* F1
Homo.	19	0.68	5	0.80	0.80	0.80	**0.87**	0.87	0.87	0.85	0.86	0.85	0.42
Poly.	39	0.36	5	0.72	0.76	0.72	0.78	0.80	0.78	**0.86**	0.86	0.87	0.37
Overall	58	0.52	10	0.76	0.78	0.76	0.82	0.83	0.82	**0.85**	0.86	0.86	0.39

Table 1: Data size and experimental results summary. *WSD* refers to the classifiers that are trained for WSD, *Cluster* for cluster assignment. *10-Cluster* classifiers refer to the classifiers that are trained for cluster assignment with only the top 10 features. *10-Rand.* refers to the classifiers that are trained for cluster assignment with random 10 features. *F1*, *Precision* and *Recall* scores are given. The best F1 score overall and for each type is given in bold.

As our data, we use the sentences of the selected senses from SemCor. We do not include the senses that have less than 10 sentences in SemCor. We balance the number of sentences for each group by random under-sampling. We exclude the words that show inherent and metonymical polysemy because different types of polysemy have different characteristics (Klepousniotou and Baum, 2007; Klepousniotou et al., 2008) and we focus only on irregular polysemy.[2]

4 Method

The goal of this study is to first, identify the sentence context features that determine the similarity of the contextualized representations. For this purpose, we identify the sentence context features that lead to the formation of clusters in word sense clustering with CWEs. Then, we evaluate these features' significance to WSD.

In order to identify these features, we conduct a cluster explainability experiment. First, we perform word sense clustering and cluster the sentences of a word (Section 4.1). As the next step, we aim to determine the sentence context features that are responsible for the formation of the clusters. For this, we try to recreate the clusters using the sentence context information of the sentences alone. We formulate this task as a classification task. We represent each sentence with a set of sentence context features and we assign the sentences to the clusters based on these features using classifiers. These classifiers are trained to predict the cluster labels from the sentence context features of the sentences (referred to as *sentence-feature classifiers*) (Section 4.2).

This gives us the advantage of representing sentences with discrete features, as opposed to contextualized representations which are continuous. This enables us to identify the specific sentence context features that contribute to the classifier performance. For this purpose, we use *recursive feature elimination* and we determine the top 10 features that are most important for the performance of the classifiers. Finally, we qualitatively evaluate the significance of the selected features to WSD (Section 4.3).

4.1 Clustering

As explained in Section 3, we focus on sense groups and each sense group contains several senses of a word. We perform word sense clustering with each sense group, clustering the sentences of senses within each group.

We cluster the sentences using the word's CWEs in these sentences. We extract the CWEs from the English BERT model (*base, cased*)[3] from each layer.[4] In cases where the words are tokenized into subwords, only the first subword's embedding is used.

We use the *K-means* clustering algorithm, selecting k as the number of senses in each group.[5] We evaluate the performance by comparing cluster labels to the sense labels using *Adjusted Rand Index* (ARI). To be able to compare the performance of each lexical ambiguity type, first, we determine the performance for each sense group within a type, then calculate their average and this average represents the performance of each lexical ambiguity type. We compare the performance change across layers and also the performances based on the best-performing layer. We expect homonymy to perform better in word sense clustering based on the

[2]See Appendix B for the list of words and the number of their senses used in this study.

[3]We choose the BERT *cased* model because it encodes more concepts relevant to WSD, such as words' WordNet concepts, compared to the *uncased* model, which encodes more linguistic concepts (Sajjad et al., 2022).

[4]We use the Transformers library (Wolf et al., 2020) for extracting the embeddings.

[5]Sci-kit learn library is used for the implementation (Pedregosa et al., 2011).

findings of the previous studies (Nair et al., 2020; Haber and Poesio, 2021).

4.2 Sentence-Feature Classifiers

We use the resulting clusters from the previous experiments for training and testing the classifiers. We use the last layer's results because this layer performs best in the clustering experiment. For each sense group, we train a classifier for cluster assignment: to predict which cluster a sentence belongs to. We only select the sense groups that have more than 25 sentences in each cluster since this experiment requires data for training. This reduces the number of sense groups we focus on in this experiment. Additionally, even though we do not limit the number of senses in each group, we end up with only two senses per group. See Table 1 for the number of sense groups for each experiment.

Our aim is to predict the cluster that each sentence belongs to based on its sentence context features. First, we represent each sentence with a manually selected sentence context features; bag-of-words, morphological properties of the target word (tense, number, etc.), POS tag of the word's neighbors, the syntactic role of the target word, and the position of the target word in the sentence. We create a sentence feature matrix by binarizing and combining features, resulting in a one-hot representation for each sentence. We select these features to be able to represent the sentences with their context as much as possible. Additionally, we aim to investigate whether the position of the word in the sentence affects the similarity of the representations, considering that positional embeddings are added to the word's representations with the BERT model.[6]

We process the sentences with the spaCy library[7] to automatically extract this information from the sentences. For the morphological properties of the target word, we use the fine-grained POS tag of the word. Similarly, we use the dependency label of the words as their syntactic role.[8] Bag-of-words representations of the sentences are created by first lemmatizing the sentences, also with spaCy.

For each sense group, we use the sentences in all clusters as our training and test data (split by 3:1). We give each cluster a label (0, 1). For each sentence, the input is its sentence feature matrix and the output is the label of its cluster. We use the linear SVM algorithm to train the classifiers because it is ideal in cases where the number of features is larger than the number of samples. Since each sense group contains two senses in this experiment, our task is to do binary classification to assign the correct cluster label.

We evaluate the performance of the classifiers based on lexical ambiguity type. We calculate the average F1 score (as well as precision and recall scores) for all sense groups within a type and consider it as each type's performance. The high performance of the classifiers will be an indication that clusters can be recreated with these features and therefore these features can explain the clusters.

Additionally, we train another type of classifiers: classifiers for WSD. These classifiers are trained similarly to the classifiers for cluster assignment; for each sense group and using sentence features as the input. But this time instead of predicting the cluster labels, the classifiers are tasked to predict the sense labels. We compare the performances of the classifiers trained for cluster assignment and WSD. This way, we aim to understand how helpful these features are for WSD to begin with. If the classifiers for cluster assignment perform better than the classifiers for WSD, this can suggest that the clusters are more distinguishable by the sentence context features than the senses and this is already an indication that clusters are formed by the sentence context features that are insignificant to WSD. Additionally, we expect these classifiers to perform better with homonymy compared to polysemy because sentence context is more helpful for the disambiguation of homonymy.

4.3 Feature Importance

In order to identify the sentence context features that are responsible for the clusters, we need to identify the features that are important for the performance of the classifiers for cluster assignment. For this purpose, we apply *recursive feature elimination* (RFE) on top of the classifiers.[9] RFE functions as a wrapper feature selection algorithm. It assesses the importance of each feature and iteratively removes the least important ones. The model is then re-fitted with the reduced feature set, and this process continues until the desired number of features is achieved.

[6] For detailed information about the size of the data and the feature matrices for each word, see Appendix D.

[7] Available at: https://spacy.io/

[8] See Appendix A for the tags used and their desciptions.

[9] Sci-kit learn library is used both for the implementation of RFE and the training of the classifiers.

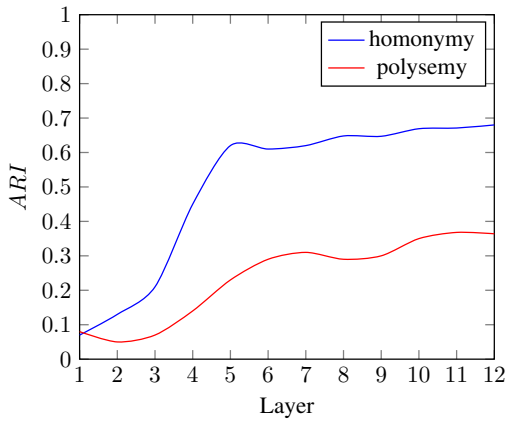

Figure 2: Layer-wise clustering performance with homonymy and polysemy.

We apply RFE to reduce the sentence feature matrix to 10 features. We train the classifiers with RFE using the same training and test datasets as the classifiers trained with the full-sentence feature matrices. By selecting the top 10 features from the sentence feature matrix, we determine which features are important for the correct classification. Then, we evaluate the performance of the classifiers that are trained only with the top 10 features on cluster assignment. Additionally, we train classifiers for cluster assignment with randomly selected 10 features for each sense group in order to establish a baseline. The baseline is determined by averaging the performance of the classifiers across 5 runs for all sense groups. The high performance of the classifiers that are trained with the top 10 features will indicate that these features are responsible for the formation of the clusters.

Finally, we assess the significance of these features to WSD for each word through qualitative evaluation. There are two reasons why we opt for qualitative evaluation. First, there might be coincidental similarities in sentence context within the sentences of one sense that help the clustering process but that are insignificant to WSD. For example, a verb's most past tense occurrences might coincidentally overlap with one sense, and generalizing over this pattern can help the WSD process. However, relying on these patterns is less than ideal. In such cases, performance-based evaluation cannot effectively capture the significance of these features because they might artificially boost performance. Our primary aim is to uncover these features. The second reason is the limited data size. Qualitative evaluation allows for a deeper understanding, even in situations where the data is limited.

5 Results

5.1 Clustering

As shown in Figure 2, the clustering performance improves across the layers, with the highest performance observed in the final layer for both types. Word sense clustering performs better with homonymy than polysemy. In the last layer, ARI score is 0.68 for homonymy and 0.36 for polysemy, as shown in Table 1. Regarding the layer-wise performance of the clustering, the pattern for homonymy and polysemy is different. There is a significant performance improvement observed between the 3rd and 5th layers for homonymy. However, there isn't that steep gain in performance for polysemy overall. This suggests that homonymous senses are mostly disambiguated early in the model layers. Overall, these results are in line with our expectations; the performance with homonymy is higher than with polysemy.

5.2 Sentence-Feature Classifiers

The classifiers for WSD achieve an F1 score of 0.80 for homonymy and 0.72 for polysemy. This difference supports our hypothesis; sentence context features are more useful for the disambiguation of homonymy than polysemy. Regarding the classifiers for cluster assignment, there are also performance differences for each lexical ambiguity type. The performance is better with homonymy (0.87) than with polysemy (0.78). Overall, they achieve an F1 score of 0.82.

The classifiers for cluster assignment show better performance compared to the classifiers for WSD, with an overall increase of 0.06 point. There is an increase for both homonymy (0.07) and polysemy (0.05). This increase suggests that the sentence context features are more prominent in the clusters than the original sense sentences and the clusters are more easily distinguishable by their features compared to the senses. Finally, the overall high performance of the classifiers for cluster assignment suggests that the selected sentence context features are a good starting point for feature selection. The results of the classifier performances can be seen in Table 1.

5.3 Feature Importance

Top 10-Feature Classifiers for Cluster Assignment. The classifiers trained with the top 10 features for cluster assignment achieve good performance with an overall F1 score of 0.85, surpassing

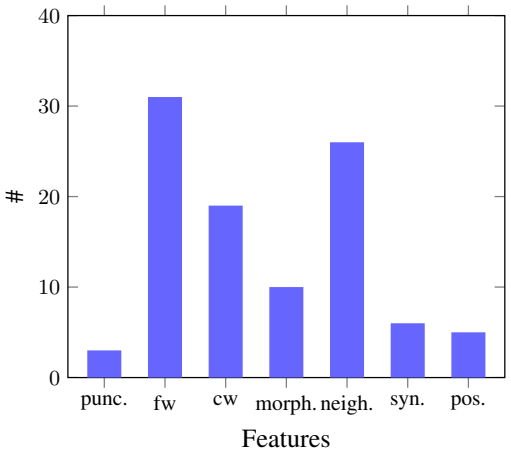

Figure 3: Count of each feature category. The categories are punctuation marks, function words (*fw*), content words (*cw*), neighboring words, the position of the target words in the sentences, and morphological properties and syntactic roles of the target words.

the random baseline (F1: 0.39) by a large margin (see Table 1). This indicates that: *the selected top 10 features are able to recreate the clusters to a great extent.*

Evaluation of the Selected Features. We group the top 10 features selected for all sense groups by their similarities and 6 categories are formed. These categories are punctuation marks, function words, content words, POS tags of neighboring words, morphological properties of the target words, syntactic roles of the target words, and positions of the target words in the sentences. Their counts can be seen in Figure 3.[10]

Punctuation marks ('-', ';', etc.), function words ('if', 'not', etc.), and content words ('river', 'bed', etc.) are the items that are found in the bag-of-words representations of the sentences and are selected as important features. This means that the fact that there are certain items in the sentence determines the decision of the classifiers.

Furthermore, morphological properties of the target word, e.g. whether the verb is in past tense or not, and the syntactic role of the target word, e.g. whether the noun is the direct object of the sentence or not, determine the decision of the classifiers. Similarly, the POS tags of neighboring words also is a determining feature. For example, whether a verb is followed by an adverb or not or whether a verb is followed by a punctuation or not. Finally, the position of the target word is also a determining feature. However, only the 6th, 7th, 8th, 9th and 10th positions are found to be important.

First, without looking at the details, it is apparent that certain feature categories lack significance or have little significance to WSD. These categories include punctuation marks, the position of the target word in the sentence, the syntactic role and the morphological properties of the target words. On the other hand, features such as POS tags of the neighboring words, and the existence of some content words in the sentence can carry more significance. For example, whether a verb is followed by a preposition or not can be a good indicator of a sense. Similarly, the presence of a word in the sentence can signal one sense, as previously shown in (1) for 'bank'.

Yet, a closer examination reveals even more striking results. *In most cases, the important features are insignificant to WSD*, except for a few words and this explains the poor clustering performance. The main issue is that most of the time, *a particular insignificant feature is found in both sense sentences and causes these sentences to cluster together*. The features from all categories affect the performance like this. For example, sentences of different senses of a verb are clustered together because, in all of them, the verb is in the past tense, as illustrated in example (3) with two senses of the verb 'indicate'.

(3) a. *be a signal for or a symptom of:*
"The statistics hardly **indicated** that..."
b. *to state or express briefly:*
"He **indicated** that requests would..."

Other times, *one feature that is not significant to WSD is found only in the sentences of one sense coincidentally and causes these sentences to cluster together*. While this might affect the performance positively, it does so for reasons that are not ideal. This finding aligns with our expectations. For example, the word 'other' is selected as an important feature for the clusters of 'time'. This feature is not significant to the WSD of this word and it is even not found in direct syntactic relation with the target word in the sentences as in example (4).

(4) The debris of his **other** careers was piled everywhere; a pile of wire cages for mice from his **time** as a geneticist and a microscope lying on its side on the window sill...

Finally, we do not observe specific patterns for dif-

[10] A detailed list can be seen in Appendix C.

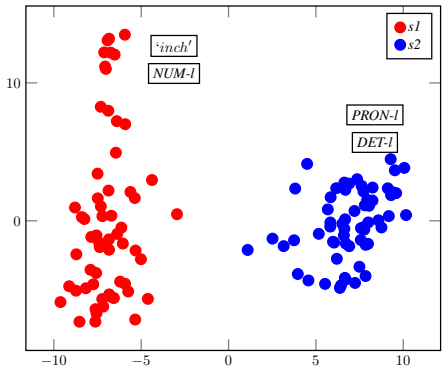

Figure 4: PCA visualization of the embeddings of 'foot' in different sense sentences and important features found in each sentence cluster of the word. The embeddings are extracted from the last layer of the BERT model. The features are 'inch', *NUM-l* (the left neighbor is a numeral), *PRON-l* (the left neighbor is a pronoun) and *DET-l* (the left neighbor is a determiner).

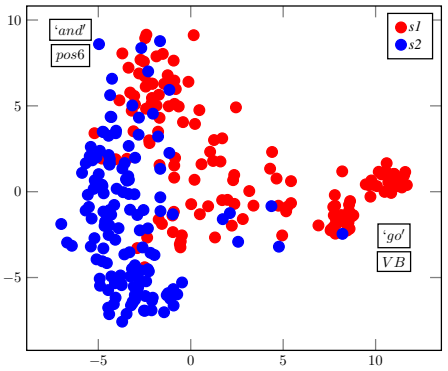

Figure 5: PCA visualization of the embeddings of 'ask' in different sense sentences and important features found in each sentence cluster of the word. The embeddings are extracted from the last layer of the BERT model. The features are 'go', 'and', *VB* (the verb is in base form), and *pos6* (the word is the 6th token in the sentence).

ferent lexical ambiguity types, however, in general, we observe that for some words, there are more clear clues in the sentence context that are helpful for disambiguation.

An Example: 'ask' vs. 'foot'. 'ask' in its first sense means "to request something" as in (5-a) and in its second sense means literally "to ask a question" and in this sense, it is also frequently used with direct speech as in (5-b). 'foot' in its first sense is the *body part* and in its second sense, it is the *measuring unit*, as illustrated in (6).

(5) 'ask':

a. *to request something:* "He **asked** her for recommendation."

b. *to ask a question:* "Don't **ask** a question.", "'Who said that?' he **asked**."

(6) 'foot':

a. *body part:* "He hit his **feet**."

b. *measuring unit:* "She is five **feet** tall."

Even though both of these words are homonymous, there are performance differences between them.[11] Word sense clustering achieves perfect performance with 'foot' (1.0) and bad performance with 'ask' (0.16). However, the sentence-feature classifiers for cluster assignment perform well with both words ('ask': 0.77, 'foot': 1). Looking at the classifier performance, we can conclude that the clusters of both words are distinguishable based on their sentence context features. However, looking at the word sense clustering performance, we can understand that these features are not equally significant to WSD for both words; they are significant in the case of 'foot', but not 'ask'.

The clusters of 'foot' represent each sense well as shown in Figure 4, and features for each cluster are also related to different senses of 'foot'. For the cluster related to the 'measuring unit' sense, the features 'inch', and *NUM-l* (the left neighbor is a numeral, as in "5 feet") are selected as important features. Whereas, for the cluster related to the 'body part' sense, the features *PRON-l* (the left neighbor is a pronoun, as in "his feet"), and the feature *DET-l* (the left neighbor is a determiner, as in "the feet") are selected. These features are indeed good indicators of these senses.

On the other hand, we do not see nicely formed clusters for 'ask' (see Figure 5) and we see that the similarity of the representations is driven by the features that are not significant to WSD. For example, all sentences in which the verb is in base form (*VB*) or the word is the 6th token (*pos6*) or the sentences that have the words 'go' or 'and' are clustered together.

Two senses of 'ask' occur in different sentence structures: the first sense occurs with prepositional objects, as in (5-a), and the second sense with direct speech as in (5-b). It is interesting to see that these distinctions are not captured by the clusters and these features do not determine the similarity of the representations. We also see that the sentence-feature classifier for WSD performs better (0.81) than the sentence-feature classifier for cluster as-

[11] See Appendix D for a performance comparison of all words in the last experiment.

signment (0.77) with 'ask'. This contrasts with the general pattern. This might indicate that the senses are actually distinguishable by their sentence context features, however, not these features but insignificant features are responsible for the formation of the clusters.

6 Discussion

In order to identify the sentence context features that are responsible for the similarity of the contextualized representations, we conducted a cluster explainability study and identified the sentence context features that lead to the formation of the clusters in word sense clustering with CWEs. Our results have shown that features from different categories determine the similarity of the representations; function words, punctuation marks, content words in the sentences, position of the target word in the sentence, neighboring words, morphological properties and the syntactic role of the target word. Our results are in line with Sajjad et al. (2022) who have shown that the CWEs encode both grammatical and semantic properties of the words and the clusters of CWEs reveal these similarities.

Furthermore, we qualitatively evaluated the identified features for each word and have shown that they are mostly insignificant to WSD. We observed that even when different senses of a word occur in diverse contexts and the sentence context provides clear clues for different senses (as in the case with 'ask'), the significant features do not determine the similarity of the representations in most cases. This contradicts our expectations. When the sentence context provides clear clues for different senses, e.g. in the case of homonymy, we expected the similarity of contextualized representations to be determined by these clues. However, this is not the case and there are other features in the sentences that are insignificant to WSD, that affect the similarity of the representations more.

Our analysis also has revealed that insignificant features affect the clustering performance negatively in several ways. Most commonly, some insignificant features occur in the sentences of both word senses and they lead these sentences to cluster together. This explains the poor clustering performance reported previously (Yenicelik et al., 2020) and also in this study. Additionally, in some cases, certain insignificant features occur only in the sentences of one sense by chance and they lead to the formation of clusters. Although these cases don't affect the performance negatively, this shows how the randomness in the data can affect the clustering performance.

In relation to the performance with different lexical ambiguity types, the findings of our study are in line with previous studies (Nair et al., 2020; Haber and Poesio, 2021; Sevastjanova et al., 2021). Clustering performs better with homonymy than polysemy. In addition to previous studies, our results have shown that homonyms are more distinguishable by sentence context features than polysemes and their disambiguation can be more easily achieved with a simple classifier trained with these features. However, contextualized representations' similarity is not consistently determined by the sense-significant features even for homonyms.

7 Conclusion

The information encoded in contextualized representations which determines their similarity is not significant to WSD in most cases. This shows that these representations do not capture the different meanings of words as expected, explaining why using CWEs of pre-trained language models alone does not yield sufficient performance in unsupervised WSD. In the future, we plan to explore possible strategies to create contextualized representations that are more suitable to WSD by limiting the information that is insignificant to WSD encoded in the representations. This way, we aim to enhance unsupervised WSD performance.

8 Acknowledgement

This study is funded by the project "Coercion and Copredication as Flexible Frame Composition" funded by DFG (Deutsche Forschungsgemeinschaft). We would like to thank the anonymous reviewer for their valuable comments. Lastly, special thanks to David Arps for interesting and stimulating discussions.

References

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.

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.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

Marcos Garcia. 2021. Exploring the representation of word meanings in context: A case study on homonymy and synonymy. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3625–3640, Online. Association for Computational Linguistics.

Luke Gessler and Nathan Schneider. 2021. BERT has uncommon sense: Similarity ranking for word sense BERTology. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 539–547, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Janosch Haber and Massimo Poesio. 2021. Patterns of polysemy and homonymy in contextualised language models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2663–2676, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ekaterini Klepousniotou. 2002. The processing of lexical ambiguity: Homonymy and polysemy in the mental lexicon. *Brain and Language*, 81(1):205–223.

Ekaterini Klepousniotou and Shari R. Baum. 2007. Disambiguating the ambiguity advantage effect in word recognition: An advantage for polysemous but not homonymous words. *Journal of Neurolinguistics*, 20(1):1–24.

Ekaterini Klepousniotou, G. Bruce Pike, Karsten Steinhauer, and Vincent Gracco. 2012. Not all ambiguous words are created equal: An eeg investigation of homonymy and polysemy. *Brain and Language*, 123(1):11–21.

Ekaterini Klepousniotou, Debra Titone, and Carolina Romero. 2008. Making sense of word senses: the comprehension of polysemy depends on sense overlap. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 34(6):1534.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Timothee Mickus, Denis Paperno, Mathieu Constant, and Kees van Deemter. 2020. What do you mean, BERT? In *Proceedings of the Society for Computation in Linguistics 2020*, pages 279–290, New York, New York. Association for Computational Linguistics.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Sathvik Nair, Mahesh Srinivasan, and Stephan Meylan. 2020. Contextualized word embeddings encode aspects of human-like word sense knowledge. In *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, pages 129–141, Online. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830.

Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Firoj Alam, Abdul Khan, and Jia Xu. 2022. Analyzing encoded concepts in transformer language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3082–3101, Seattle, United States. Association for Computational Linguistics.

Rita Sevastjanova, Aikaterini-Lida Kalouli, Christin Beck, Hanna Schäfer, and Mennatallah El-Assady. 2021. Explaining contextualization in language models using visual analytics. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 464–476, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame,

Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

David Yenicelik, Florian Schmidt, and Yannic Kilcher. 2020. How does BERT capture semantics? a closer look at polysemous words. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 156–162, Online. Association for Computational Linguistics.

Tag	Description
nsubj	nominal subject
pobj	object of a preposition
attr	attribution
xcomp	open clausal complement
npadvmod	noun phrase as adverbial modifier

Table 2: Dependency labels used in this study, from spaCy model *en_core_web_trf*.

Tag	Description
NN	Noun, singular or mass
NNS	Noun, plural
VBD	Verb, past tense
VBP	Verb, non-3rd person singular present
VBZ	Verb, 3rd person singular present
VB	Verb, base form

Table 3: Fine-grained POS tags used in this study, from Penn Tree Bank (Marcus et al., 1993).

Tag	Description
ADP	adposition
PUNCT	punctuation
PART	particle
SCONJ	subordinating conjunction
PRON	pronoun
DET	determiner
ADV	adverb
NUM	numeral

Table 4: POS tags used in this study, from Universal Dependencies (Nivre et al., 2016).

A Feature Tags

The feature tags for the morphological properties of the words, the syntactic role of the words, and the POS tags of neighboring words that are used in this study can be found in Table 2, 3, 4. For the morphological properties of the words, we use their fine-grained POS tag (Table 3). For the syntactic role of the words, we use their dependency label (Table 2). All the labels are obtained by processing the sentences with spaCy.

ask: 1	heart: 2	**life: 5**
begin: 3	produce: 2	point: 5
degree: 3	put: 3	raise: 3
drive: 3	table: 2	**time: 2**
foot: 2	right: 2	**way: 4**
heart: 2	case: 3	**world: 4**
indicate: 2	consider: 4	plane: 2
light: 3	cover: 2	lead: 7
man: 3	**door: 2**	

Table 5: The words that are found in our dataset with their sense counts. All the words are used in the word sense clustering and the bold words are used in the cluster explainability experiment.

B Selected Words

A list of the words that are found in our dataset can be seen in Table 5. All the words and senses are used in the word sense clustering experiment. Only 10 words and 2 sense each are used in the cluster explainability experiment.

C Selected Features List

A detailed list of selected features from each category can be seen in Table 6.

D Performance with Individual Words

The individual performance of each word can be seen in Table 7. Only the performances of the words that are used in the last experiment are reported. Both clustering performance and the performance of the sentence-feature classifiers are reported. The data size (number of sentences) of each word can be also seen in Table 7.

Feature Category	Features
Neighbouring Word (right)	ADP: 3, PUNCT: 2, PART: 2, SCONJ: 1, PRON: 1, DET: 1, ADV: 1
Neighbouring Word (left)	DET: 4, PART: 2, PRON: 2, NUM: 2, ADP: 2, ADJ: 2, PUNCT: 1
Punctuation	'.': 2, '-': 1,
Function Word	'and': 1, 'after': 1, 'can': 1, 'she': 1, 'might': 1, 'the': 3, 'to': 2, 'when': 1, 'my': 1, 'on': 1, 'through': 1, 'no': 1, 'with': 1, 'just': 1, 'a': 1, 'by': 1, 'however': 1, 'in': 1, 'or': 1, ''s': 1, 'each': 2, 'which': 1, 'her': 1 'other': 1, 'their': 1, 'once': 1, 'such': 1
Content Word	'mean': 1, 'nothing': 1, 'high': 1, 'outside': 1, 'see': 1, 'first': 1, 'route': 1, 'Spencer': 1, 'new': 1, 'plan': 1, 'event': 1, 'God': 1, 'three': 1, 'hand': 1, 'go': 1, 'take': 1, 'inch': 1, 'age': 1, 'feel': 1
Syntactic Role	nsubj: 2, pobj: 1, xcomp: 1, attr: 1, npadvmod: 1
Morphological Properties	NNS: 3, VBD: 2, NN: 2, VBP:1, VBZ: 1, VB: 1
Word Position	6th: 1, 7th: 1, 8th: 1, 9th: 1, 10th: 1

Table 6: Selected important features in each category and their counts. POS tags of the word's neighbors are given for neighboring words, the dependency label of the word is given for the syntactic role, and the fine-grained POS tag is given for the word's morphological properties. See Appendix A for the descriptions of the tags used.

Word	Data#	**Clustering Performance**	Feature#	**Sentence-Feature Classifiers**								
				WSD			*Cluster*			*10-Cluster*		
				F1	P	R	F1	P	R	F1	P	R
Homonymy												
ask	278	0.16	1668	**0.81**	0.82	0.81	0.77	0.77	0.77	0.77	0.77	0.77
begin	90	0.34	1007	0.87	0.87	0.87	**0.91**	0.93	0.92	0.80	0.80	0.80
foot	119	1	1008	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
indicate	70	0.12	809	0.62	0.62	0.62	**0.85**	0.86	0.85	**0.85**	0.89	0.85
man	92	0.54	932	0.71	0.71	0.70	0.82	0.83	0.82	**0.85**	0.86	0.85
Polysemy												
life	74	0.34	775	0.59	0.60	0.60	0.81	0.83	0.81	**0.86**	0.86	0.86
man	54	0.16	690	0.66	0.81	0.68	0.84	0.84	0.84	**0.94**	0.95	0.94
time	56	0.69	762	0.71	0.76	0.71	0.84	0.84	0.84	**0.94**	0.95	0.94
way	110	0.55	1179	0.81	0.81	0.81	**0.86**	0.86	0.86	0.73	0.74	0.73
world	56	0.35	692	**0.84**	0.84	0.84	0.53	0.66	0.55	0.83	0.88	0.83

Table 7: Data size and performance details of individual words. *Data#* refers to the number of sentences. Clustering performance is evaluated using ARI. *Feature#* is the size of the sentence feature matrix. *WSD* refers to the classifiers trained for WSD, *Cluster* for cluster assignment. *10-Cluster* classifiers refer to the classifiers trained for cluster assignment with only the top 10 features. *F1*, *Precision* and *Recall* scores are given for each classifier. The best F1 score for each word is given in bold.