Context-Aware Semantic Similarity Measurement for Unsupervised Word Sense Disambiguation

Jorge Martinez-Gil

Software Competence Center Hagenberg GmbH
Softwarepark 32a, 4232 Hagenberg, Austria
jorge.martinez-gil@scch.at

Abstract

The issue of word sense ambiguity poses a significant challenge in natural language processing due to the scarcity of annotated data to feed machine learning models to face the challenge. Therefore, unsupervised word sense disambiguation methods have been developed to overcome that challenge without relying on annotated data. This research proposes a new context-aware approach to unsupervised word sense disambiguation, which provides a flexible mechanism for incorporating contextual information into the similarity measurement process. We experiment with a popular benchmark dataset to evaluate the proposed strategy and compare its performance with state-of-the-art unsupervised word sense disambiguation techniques. The experimental results indicate that our approach substantially enhances disambiguation accuracy and surpasses the performance of several existing techniques. Our findings underscore the significance of integrating contextual information in semantic similarity measurements to manage word sense ambiguity in unsupervised scenarios effectively.

Keywords: Natural Language Processing, Knowledge Engineering, Semantic Similarity Measurement

1. Introduction

Semantic similarity refers to the extent to which two text pieces convey the same meaning [24]. Traditional semantic similarity measurement strategies rely on various approaches considering the overlap of features between the two texts to be compared [12]. However, these approaches suffer from several limitations, as they fail to consider the context in which the words and phrases are used. In other words, in conventional semantic similarity measures, the likeness between two entities is based on their definitions, relationships, and other linguistic or extrinsic features [16]. However, in real-world applications, the context in which entities are being compared can influence their semantic similarity.

Furthermore, word sense ambiguity is a common problem in natural language processing (NLP) because words often have multiple meanings depending on their context. Word sense disambiguation (WSD) aims to identify the correct meaning of a word in a given context [23]. While supervised WSD approaches have achieved high accuracy, they are limited by the availability of annotated data. In contrast, unsupervised approaches rely on something other than annotated data but often suffer from lower accuracy due to that lack of supervision.

This research proposes a Context-Aware Semantic Similarity (CASS) measurement approach for unsupervised WSD to overcome the accuracy of the results traditionally achieved through unsupervised strategies. The method incorporates contextual information into the similarity measurement process to reduce language ambiguity. This approach allows us to automatically identify the most probable sense of an ambiguous word based on its context without relying on annotated data. By considering the context in which words are being compared, CASS can improve the accuracy and relevance of search results and recommendations. This can be particularly useful in many domains where relevance may vary based on user preferences and history.

We must also remark that our strategy is not the first in that direction since several unsupervised WSD techniques have been proposed [29, 22, 8, 27]. However, these methods consider the specific context in which the words appear following a different approach. Our proposed strategy addresses currently limitations by means of a novel CASS that adequately incorporates contextual information into the disambiguation process. Therefore, the primary contributions of this research can be summarized as follows:

- A novel approach to NLP that considers the context in which words are used to improve the accuracy and relevance of language models beyond the traditional methods to identify synonyms. The proposed approach for unsupervised WSD can benefit languages with limited annotated data, whereas supervised approaches may not be as effective.

- Evaluation of the proposed method on a complete benchmark dataset and comparison of its performance with several state-of-the-art unsupervised WSD techniques. The experimental results show that the method improves disambiguation accuracy and outperforms several existing techniques, especially when annotated data is limited or unavailable.

The remainder of this paper is organized as follows. Section 2 provides an overview of related work in unsupervised WSD. Section 3 introduces the problem statement of this research. Section 4 presents the details of the proposed CASS measurement approach. In Section 5, the experimental setup is described, and the evaluation results are presented. Section 6 discusses the results obtained and directions for future work. Finally, the paper concludes in Section 7.

2. State-of-the-art

This section presents an overview of CASS measurement, discussing its objectives, and methods. We review the state-of-the-art techniques used for this task and discuss their challenges and limitations, such as the difficulty of capturing context-dependent nuances and the lack of a universally accepted evaluation methodology. Finally, we highlight some of the potential applications.

2.1. Semantic Similarity

Semantic similarity is an essential concept in NLP that has been extensively studied in the literature [7, 11, 9, 30, 13, 18]. Traditional approaches to measuring semantic similarity are typically based on the study of inherent characteristics of the words (lexical methods) orits distribution in sufficiently meaningful text corpora (distributional semantics) [19]. Lexical methods rely on the meaning of individual words and their relationships to each other [17]. In contrast, distributional semantics techniques aim to capture the meaning of words based on their co-occurrence patterns in large corpora [2].

CASS is a family of NLP techniques that measures the semantic similarity between two words or phrases in a given context. This family has gained increasing attention in recent years due to its ability to capture not just the meaning of words but additional nuances by considering the context in which they are used [26]. It represents an extension of traditional semantic similarity measurement since the latter does not consider the text's context. Recent advances have seen the development of CASS methods that consider the context in which words are used. These approaches recognize that the meaning of a word can vary depending on the specific context in which it appears.

In addition, there is currently no universally accepted evaluation methodology for assessing the performance of CASS measures. This lack of consensus makes it difficult to compare and pick the best approaches for a variety of applications, which slows down overall progress in the field. In the research that has been done, numerous evaluation measures have been suggested. However, they frequently have drawbacks, such as favoring particular kinds of data or failing to consider the activity's total complexity. Therefore, additional research is required in order to develop a methodology for evaluation that is both comprehensive and objective.

2.2. Applications

By considering the setting in which words are employed, it is possible to implement specialized CASS measures to increase text understanding capabilities. This can be especially helpful in activities such as, for instance, sentiment analysis, in which the meaning of a word can change based on the context in which it is being used.

CASS is crucial in various applications, including web search, document classification, question-answering, and text summarization. It measures the semantic similarity between two words, and context plays a vital role in determining this similarity. Search engines use semantic similarity to retrieve documents that match the meaning of the user's query. In document classification, understanding the document's meaning is essential, while understanding the question's meaning is crucial in question-answering systems. Text summarization involves condensing much text into a shorter version, retaining essential information. Semantic similarity can enhance accuracy and relevance in these applications by capturing language nuances and context.

2.3. Word Sense Disambiguation

CASS measurement and WSD are related concepts in the field of NLP, but they differ significantly. Semantic similarity measurement involves determining how similar two words or phrases are in terms of meaning [4]. CASS measurement considers the context in which the words or phrases appear in a sentence or document and their inherent semantic properties. This approach can help capture nuances and subtleties in meaning that might be missed by other methods that rely solely on the intrinsic properties of the words or phrases.

WSD, conversely, is determining which word's meaning is intended in a particular context [1]. This is particularly important for words with multiple meanings [14]. WSD can be a challenging problem, especially when the context is ambiguous or there are few clues to help distinguish among the possible senses [6]. The challenge of unsupervised WSD is also important, as evidenced by a number of recent papers that provide ideas for meeting the challenge when adequate training datasets are not available [29, 22, 8, 27]. Therefore, CASS and WSD are essential tools in NLP, but they have different goals and use cases.

2.4. Contribution over the state-of-the-art

The training-test gap for models based on unsupervised language modeling makes it difficult for these methods to both compute semantic similarity and perform word sense disambiguation correctly. Existing annotated datasets are typically small, making it challenging to train supervised neural models. Our proposed strategy, which incorporates CASS, is the foundation of our contribution to the state-of-the-art since it alleviates the problem in scenarios where appropriate training datasets are unavailable. We have tested our strategy against the most recent WSD benchmark dataset and discovered that it performs better in terms of accuracy than other methods. Furthermore, our approach is appropriate for large-scale applications and information retrieval because it is computationally effective and scalable. Our work thus advances the development of CASS methodologies and shows how context integration can increase the accuracy and robustness of WSD techniques.

3. Problem Statement

There are several methods for calculating CASS. Each method has its strengths and limitations, and the choice of method depends on the application's specific requirements. However, we can decompose the general problem into several subproblems and define them formally, as we will see below.

3.1. Context-Aware Semantic Similarity Measurement

The problem of context-aware semantic similarity measurement (CASS) can be formulated mathematically as follows:

Let $\mathcal{C}$ be the set of contexts, $\mathcal{W}$ be the set of words, and $\mathcal{S}$ be the set of semantic similarity scores between word pairs.

Given a context $c \in \mathcal{C}$ and two words $w_1, w_2 \in \mathcal{W}$, the task is to compute a CASS score $\mathcal{S}(c, w_1, w_2) \in \mathcal{S}$ between the word pair w_1, w_2 in the context c.

This can be formally developed as a mathematical expression as in Eq. 1:

$$\mathcal{S}(c, w_1, w_2) = f(c, w_1, w_2) \tag{1}$$

where f is the function that maps a context c and two words w_1 and w_2 to a semantic similarity score.

The goal is to find a function f that considers the context c in computing the semantic similarity score $\mathcal{S}(c, w_1, w_2)$. This challenge might benefit from incorporating contextual information, such as the topic or domain of the text, the user's interests, or the social context, into the calculation of semantic similarity.

3.2. Word Sense Disambiguation

Let $\mathcal{W}$ be a set of words with multiple senses, and let $\mathcal{S}$ be a set of senses associated with each word in $\mathcal{W}$. Let $\mathcal{C}$ be a corpus of text consisting of a set of documents $\mathcal{D} = \{d_1, d_2, ..., d_n\}$. For each word w in $\mathcal{W}$, let $\mathcal{T}(w)$ be the set of occurrences of w in $\mathcal{C}$, and let $\mathcal{S}(w)$ be the set of senses associated with w.

The goal of WSD is to assign a single sense s in $\mathcal{S}(w)$ to each occurrence t in $\mathcal{T}(w)$, such that the assigned sense is the most appropriate for the context in which t appears.

Formally, let $\mathcal{S}'(w) = \{s_1, s_2, ..., s_m\}$ be a set of candidate senses for word w. For each occurrence t in $\mathcal{T}(w)$, we seek to find the sense s in $\mathcal{S}'(w)$ that maximizes the probability $\mathcal{P}(s|t, C)$, where $\mathcal{C}$ is the context in which t appears. This can be expressed as in Eq. 2.

$$s* = argmax_s \mathcal{P}(s|t, C) \tag{2}$$

where $s*$ is the assigned sense for t, and $argmax_s$ denotes the sense that maximizes the probability.

Several approaches to estimating the probability $\mathcal{P}(s|t, C)$ include supervised learning, unsupervised learning, and knowledge-based methods. In supervised learning, an annotated dataset of word occurrences with their corresponding senses is used to train a classifier that predicts the sense for new occurrences. In unsupervised learning, clustering or probabilistic models usually group similar word occurrences into clusters, each representing a sense. In knowledge-based methods, external knowledge sources, such as dictionaries or semantic networks, are used to infer the most appropriate sense for a given word occurrence.

4. Methods

In our research, we aim to tackle the challenge of word ambiguity, which refers to the problem of words having multiple meanings based on the context in which they are used. We propose adapting existing methods incorporating contextual information to address this issue.

Among the current state-of-the-art methods for contextual language processing, we identified four stand-out approaches: BERT [5], ELMo [25], USE [3], and WMD [10]. Each method employs a unique technique for capturing contextual information, making them suitable for different use cases [20].

To adapt these methods for addressing the issue of word ambiguity, we propose using them to create contextualized embeddings. This involves representing each text unit as a vector that considers the context in which it appears. For instance, BERT (Bidirectional Encoder Representations from Transformers) [5] is a deep neural network that uses a transformer architecture to generate contextualized word embeddings. Similarly, ELMo (Embeddings from Language Models) [25] creates embeddings by training bidirectional models on large text corpora.

On the other hand, USE (Universal Sentence Encoder) [3] is a pre-trained encoder that can be used to generate sentence embeddings that capture the contextual meaning of a sentence. Lastly, WMD (Word Mover's Distance) [10] is a distance-based metric that calculates the similarity between two documents based on the distance between their constituent words.

Adapting these methods to capture contextual information and create contextualized word embeddings allows improving the performance of tasks that involve disambiguating words based on context. Our contribution is to show how existing methods can be adapted for dealing with the problem of word ambiguity, which has important implications for a wide range of applications.

4.1. Generic approach

We have two documents, $\mathcal{X}$ and $\mathcal{Y}$, and we want to calculate their semantic similarity using embeddings. We first obtain the embeddings for each word in the documents denoted by $\mathbf{w}\mathcal{X}, i$ and $\mathbf{w}\mathcal{Y}, j$, where i and j index the words in the documents $\mathcal{X}$ and $\mathcal{Y}$, respectively.

One common method to calculate the semantic similarity between the two documents is to use the average word embeddings in each document [21]. This means that the document embeddings can be obtained by averaging the word embeddings as follows:

$\mathbf{e}_{\mathcal{X}} = \frac{1}{n_{\mathcal{X}}} \sum_{i=1}^{n_{\mathcal{X}}} \mathbf{w}_{\mathcal{X},i}$
$\mathbf{e}_{\mathbf{B}} = \frac{1}{n_{\mathcal{Y}}} \sum_{j=1}^{n_{\mathcal{Y}}} \mathbf{w}_{\mathcal{Y},j}$

where $n_{\mathcal{X}}$ and n_B are the number of words in the documents $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Once we have the document embeddings $\mathbf{e}_{\mathcal{X}}$ and $\mathbf{e}_{\mathcal{Y}}$, we can calculate their semantic similarity ss using the cosine approach as in Eq. 3:

$$\text{ss} = \frac{\sum_{i=1}^{n_{\mathcal{X}}} \sum_{j=1}^{n_{\mathcal{Y}}} \mathbf{w}\mathcal{X}, i \cdot \mathbf{w}\mathcal{Y}, j}{\sqrt{\sum_{i=1}^{n_{\mathcal{X}}} \|\mathbf{w}\mathcal{X}, i\|^2} \cdot \sqrt{\sum j = 1^{n_{\mathcal{Y}}} \|\mathbf{w}_{\mathcal{Y},j}\|^2}} \tag{3}$$

Where:

$\|\cdot\|$ denotes the L2 norm of the embeddings, i.e., the length of the embedding vectors, and $\cdot$ denotes the dot product between the embeddings

The sense with the highest similarity score is selected as the disambiguated sense for the target word. However, the most challenging part remains: figuring out which embeddings yield the best results.

4.2. BERT embeddings

One popular method that could be used for CASS is based on BERT embeddings [5]. BERT embeddings are vector representations of words or sentences in a high-dimensional space learned from large corpora of text. BERT embeddings are context-aware since they capture the meaning of text units based on their surrounding context and can be used to measure semantic similarity.

Let $x_1, x_2, ..., x_n$ be a sequence of input tokens representing a sentence, and let h_i be the contextualized representation for the i-th token obtained using the BERT model.

We can obtain the sentence-level embedding S by taking a weighted average of the token embeddings as in Eq. 4:

$$S = \frac{1}{n} \sum_{i=1}^{n} \alpha_i h_i \tag{4}$$

where α_i is the weight assigned to the i-th token, and is given by Eq. 5:

$$\alpha_i = \frac{\exp(w^T h_i)}{\sum_{j=1}^{n} \exp(w^T h_j)} \tag{5}$$

Here, w is a learnable parameter vector that determines the importance of each token in the sentence representation.

Note that the weights α_i are learned during training and are used to give higher importance to tokens more relevant to the task. In practice, the sentence-level embedding S is often used as input to other NLP tasks.

4.3. ELMo

ELMo is a deep contextualized word representation model using a bi-directional language (biLM) to generate word embeddings [25]. The biLM is trained on a large corpus of text data to predict the next word in a sequence of words given the previous words in both forward and backward directions.

By combining the hidden states of the biLM at each layer, one can produce the ELMo representation of a word. Let us denote the biLM as a function $f_{biLM}(x)$ that takes a sequence of words x as input and produces a set of hidden states $H = h_1, h_2, ..., h_L$ at each layer l.

The ELMo representation of a sentence s_i is then computed as a weighted sum of the hidden states at each layer L as in Eq. 6:

$$ELMo^s = \gamma^s \left[\sum j = 0^{L-1} s_j \cdot \mathbf{w} j\right] + \gamma^s x \left[\sum_{j=0}^{L-1} \sum_{k=1}^{T_j} s_{j,k} \cdot \mathbf{w}_{j,k}\right] \tag{6}$$

where ELMo represents the sentence embedding for a given sentence s, L is the number of layers in the ELMo model, Tj is the number of tokens in the j-th layer, s_j and $s_{j,k}$ are the activations of the j-th layer for the entire sentence and the k-th token in the j-th layer, respectively, $\mathbf{w}j$ and $\mathbf{w}j, k$ are the corresponding weights for the j-th layer and the k-th token in the j-th layer, and γ^s and γ_x^s are scalar weights that are learned during training.

The learned weights capture the importance of each layer for the specific task and allow ELMo to generate context-dependent embeddings that are useful for our purposes.

4.4. Universal Sentence Enconder

Let us say we have two documents, $\mathcal{X}$ and $\mathcal{Y}$, and we want to calculate their semantic similarity using the USE embeddings [3]. We first obtain the USE embeddings of $\mathcal{X}$ and $\mathcal{Y}$, denoted by $\mathbf{e_X}$ and $\mathbf{e_Y}$, respectively.

The semantic similarity ss between $\mathbf{e_X}$ and $\mathbf{e_Y}$ is then calculated as in Eq. 7:

$$\mathrm{ss} = \frac{\mathbf{e}_{\mathcal{X}} \cdot \mathbf{e}_{\mathcal{Y}}}{\|\mathbf{e}_{\mathcal{X}}\| \cdot \|\mathbf{e}_{\mathcal{Y}}\|} \tag{7}$$

Where:

$\cdot$ denotes the dot product between the embeddings, and $\|\cdot\|$ denotes the L2 norm of the embeddings, i.e., the length of the embedding vectors

If necessary, Eq. 8 can also work with items from the documents.

$$\mathrm{ss} = \frac{\sum_i (e_{\mathcal{X}i} \cdot e_{\mathcal{Y}\rangle})}{\sqrt{\sum_i (e_{\mathcal{X}i}^2)} \cdot \sqrt{\sum_i (e_{\mathcal{Y}i}^2)}} \tag{8}$$

Where:

e_{Xi} and e_{Yi} are the i^{th} elements of the embeddings $\mathbf{e_X}$ and $\mathbf{e_Y}$, respectively.

4.5. Word Mover's Distance

The Word Mover's Distance (WMD) measures the semantic similarity between two texts, which considers the distances between the individual words in the texts [10]. The mathematical formulation of the WMD can be described as follows:

Let $\mathcal{D}$ be a metric space of word embeddings, and let be $\mathcal{X}$ and $\mathcal{Y}$ two documents consisting of n and m words, respectively. We also have a matrix T, which tells us how much of a word in $\mathcal{X}$ moves to a word in $\mathcal{Y}$, and this is represented by a non-negative number in T_{ij}. The cost of moving from one word to another is represented by $c(i,j)$, which is the distance between the word i and word j. We need to make sure that the total flow from each word in $\mathcal{X}$ is equivalent to the value of $\mathcal{X}_i$, which can be achieved by setting $\sum_j T_{ij} = \mathcal{X}_i$. With these constraints in mind, we can use Eq. 9 to find the minimum cumulative cost of transforming $\mathcal{X}$ into $\mathcal{Y}$.

$$\begin{gathered} arg\ min \sum_{i,j=1}^{n} T_{ij} c(i,j) \\ subject\ to \sum_{j=1}^{n} T_{ij} = \mathcal{X}_i\ \forall i \in \{1,2,3 \cdots n\} \ \wedge \ \sum_{i=1}^{n} T_{ij} = \mathcal{Y}_j\ \forall j \in \{1,2,3 \cdots n\} \end{gathered} \tag{9}$$

The embeddings use here are usually those from word2vec [21]. Furthermore, the optimization problem can be solved using linear programming techniques. The resulting WMD measures the semantic similarity between $\mathcal{X}$ and $\mathcal{Y}$, considering the distances between the individual words in the documents. The WMD has been shown to outperform traditional bag-of-words and vector space models in capturing the semantic similarity between texts, particularly in cases where the meaning of the words is essential [28].

5. Results

Here, we showcase the results of our WSD experiments. Through a thorough analysis of various embedding models, we have compared the outcomes of our proposed strategy with commonly employed techniques to gauge their impact.

5.1. Empirical Setup and Baseline Selection

Our research proposes a CASS measurement method for unsupervised WSD. The proposed method measures the semantic similarity between a target word and its candidate senses based on the context in which the target word appears. We compare our approach with several unsupervised techniques for each use case in the dataset. The experiments are tested on a standard computer with 32 GB of RAM memory and an i7-8700 CPU running at 3.20 GHz on Windows 10.

We will use two baselines here, one weak and one strong. The weak baseline (*Random Option*, RO) calculates the probability of giving a correct answer randomly. So in cases where two possible alternatives are considered, it would be a probability of 50%, in case of three, 33.33%, and so on. The strong baseline is one of the most commonly used baselines for WSD; the *Most Frequent Sense* (MFS) method. The MFS baseline assigns the most frequent sense of a word in a given dataset to all instances of that word. It is a method that is difficult to replicate in the real world by a computer because it requires external knowledge (i.e., the most frequent sense for a given word). However, it is a natural solution for people.

We must compute the most frequent sense of each target word in the training data to implement this strong baseline. Then, we assign the most frequent sense as the predicted sense for each instance of the target word in the test data. While this strong is very simple, the literature shows that it can be surprisingly effective, especially for words with a highly dominant sense.

5.2. Dataset

In this work, we are working with the CoarseWSD-20 dataset [15] that is a dataset for figuring out the actual meaning of words that, in practice, can have different meanings. The dataset is made from Wikipedia and only includes nouns. It focuses on 20 words that can have 2 to 5 different meanings. In total, the dataset contains 10,196 cases, and it is useful for testing WSD models, as it has all the senses in the test sets. This makes it particularly suitable for evaluating WSD models.

However, there are certain peculiarities as well. As our method is completely unsupervised, we do not need to use the training samples offered. At the same time, if we were to compete with solutions that use such training samples, the comparison would be unfair. So we will limit ourselves to the comparison with other unsupervised techniques, particularly the weak (RO) and the strong (MFS) baselines. In addition, we need to make a small adaptation in some mapping classes, to facilitate the disambiguation.

5.3. Evaluation Criteria

We use a standard evaluation metric called accuracy to evaluate the performance of different WSD models on the CoarseWSD-20 dataset. Accuracy is the proportion of correctly identified senses from the total number of instances in the test set. In addition, we will break down the results by use case and globally. Both for our approach and for the baselines we compare with.

5.4. Empirical Evaluation

We aim to evaluate the proposed CASS measurement method for unsupervised WSD and compare it with several unsupervised WSD methods. The solid blue color represents the results obtained through our strategy. The black color represents the weak baseline (the results could be replicated by selecting a random option). In contrast, the red represents the strong baseline (the results could be replicated with external knowledge about the most frequently used sense).

Strategy	Hits	Accuracy
UWSD+BERT	7,927	77.74%
MFS-Baseline	7,487	73.43%
UWSD+ELMo	7,010	68.75%
UWSD+USE	6,396	62.73%
UWSD+WMD	5,868	57.55%
RO-Baseline	4,459	43.73%

Table 1: Summary of the global results obtained using the different embedding approaches

Figure 1 shows the initial results obtained with the solution implemented by BERT. As can be seen, the results are pretty good since the weak baseline is consistently outperformed, and the strong baseline is almost always outperformed. In addition, there are many use cases where a wide margin beats the strong baseline. Considering that this strategy does not use any external resources or training, this can be considered a good performance.

Figure 2 shows the results obtained with the solution implemented by ELMo. As can be seen, this approach is better than the weak baseline but often fails to outperform the strong baseline. Therefore, the results are not optimal.

Figure 3 shows the results obtained with the solution implemented by USE embeddings. As can be seen, several results are even lower than those from the weak baseline, and the strong baseline is only surpassed in limited cases. In general, when compared to the other approaches studied, UWSD-USE is not among the best.

Figure 4 shows the results obtained with the solution implemented by WMD. As can be seen, the results are far from optimal. There are several occasions in which they are even below the weak baseline, being very rare in the cases in which they manage to overcome the strong baseline. Generally speaking, this approach is the one that yields the worst results among those studied.

5.5. Comparison with existing techniques

Table 1 summarizes all our global results. The CoarseWSD-20 dataset is still relatively young and specially designed to perform machine learning, so we are unaware of any other published work on the unsupervised WSD task. However, our experimentation has many alternatives that have been tested. As can be seen, all the proposed strategies can outperform the RO baseline (weak baseline). However, only the strategy that uses BERT embeddings can outperform the MFS baseline (strong baseline) as well. The strategy followed with implementing BERT embeddings is an excellent result since it can outperform a method that uses external knowledge without any extra knowledge or training phase.

Table 2 summarizes all the global results obtained with different BERT models. All these models have been extensively evaluated for their quality of embedded sentences. As can be seen, not all models lead to results superior to the baseline. A more in-depth analysis of the reasons why some models rank better than others remains a future work in progress.

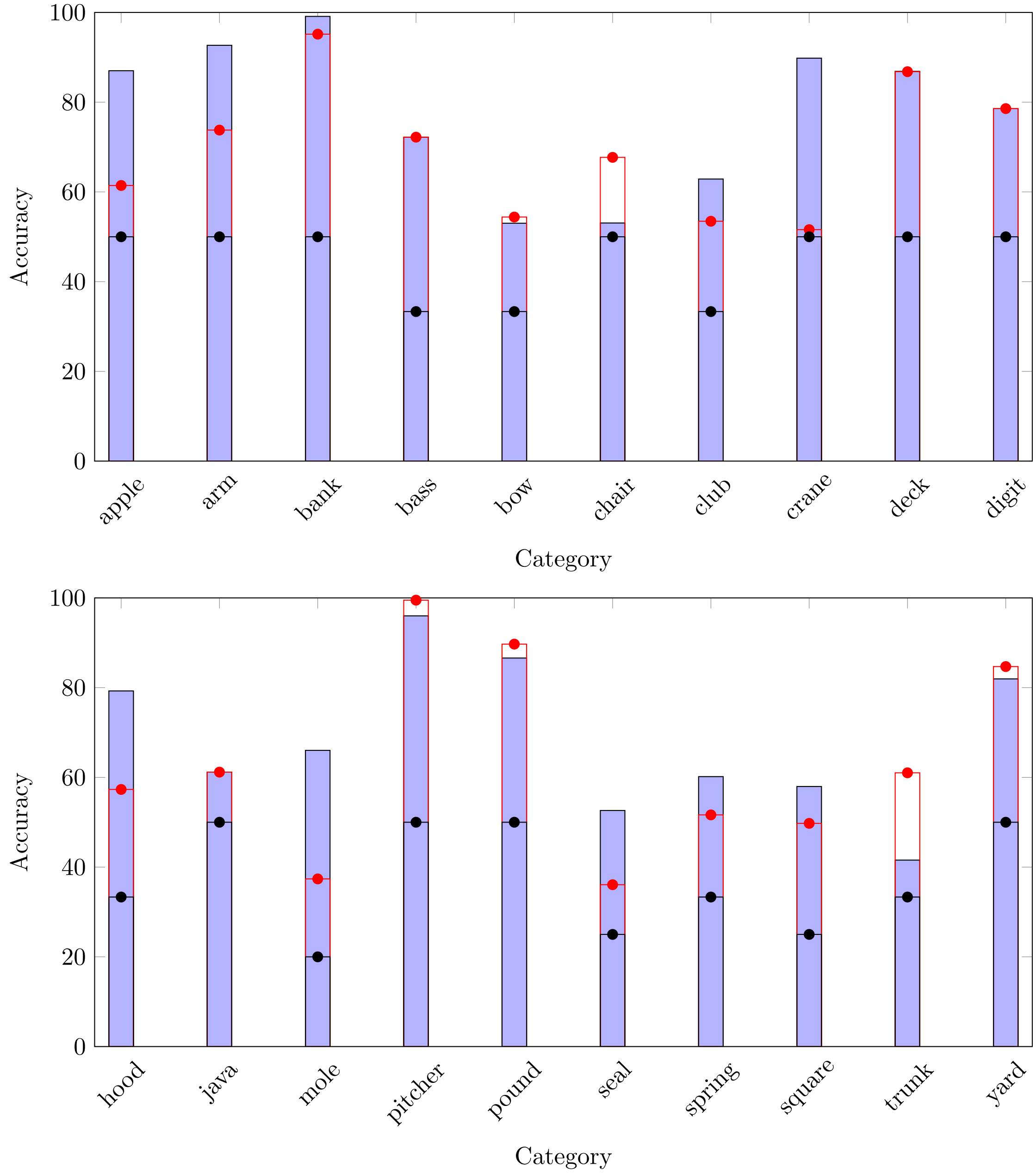

Figure 1: Results obtained for the CoarseWSD-20 dataset using UWSD+BERT. The solid blue bar represents the results obtained. While the black and red colors represent the weak and strong baselines, respectively

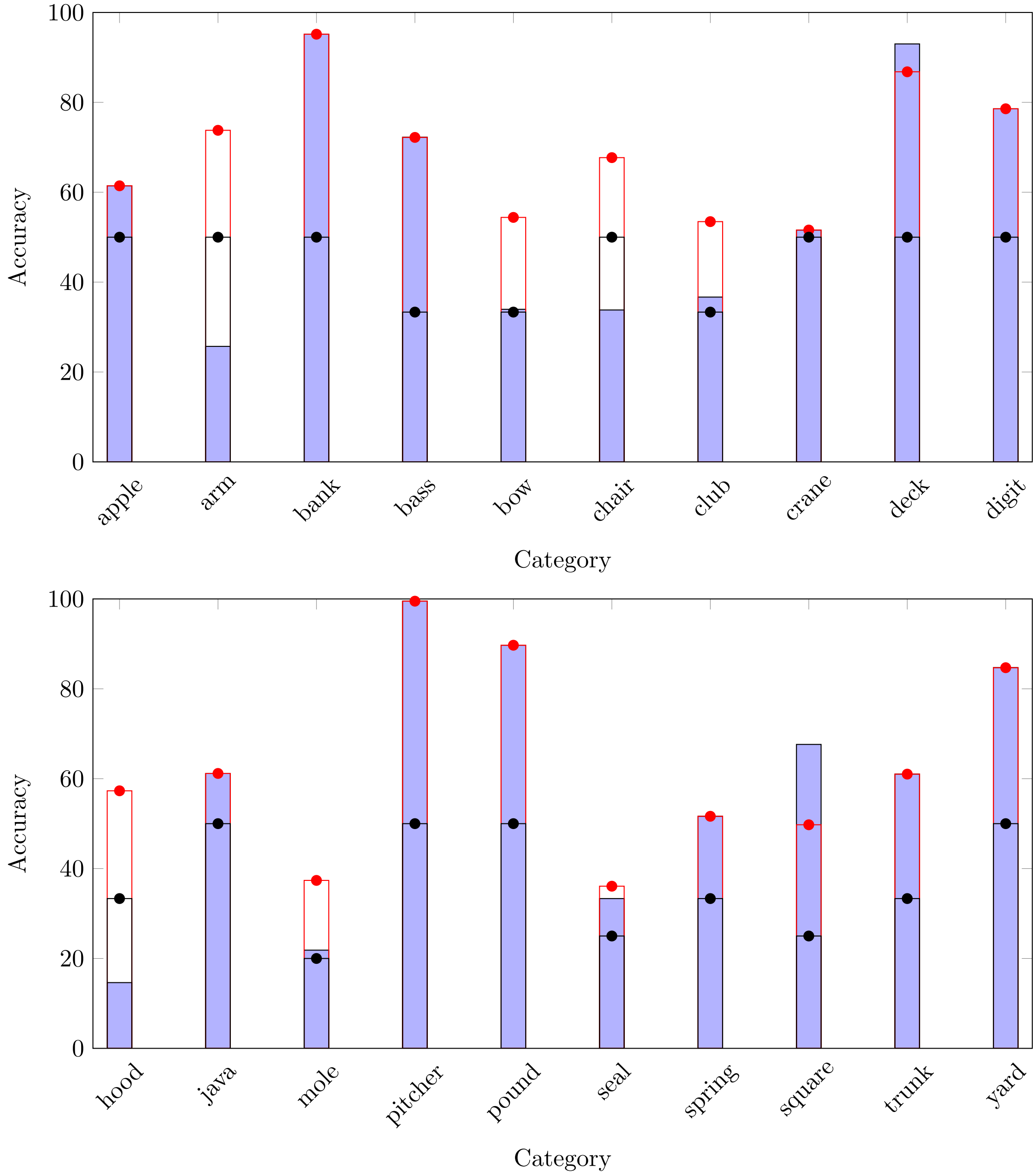

Figure 2: Results obtained for the CoarseWSD-20 dataset using UWSD+ELMo. The solid blue bar represents the results obtained. While the black and red colors represent the weak and strong baselines, respectively

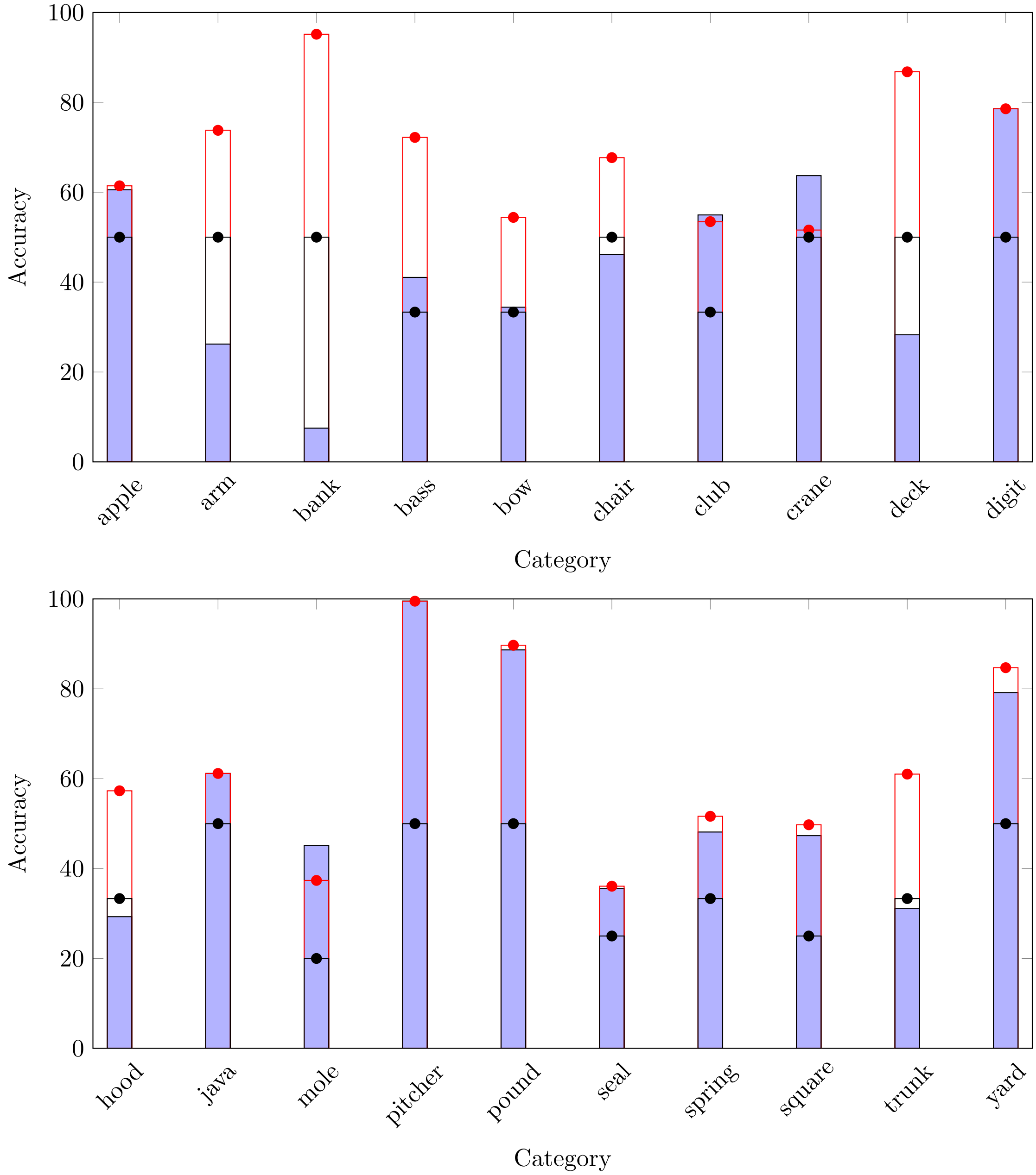

Figure 3: Results obtained for the CoarseWSD-20 dataset using UWSD+USE. The solid blue bar represents the results obtained. While the black and red colors represent the weak and strong baselines, respectively

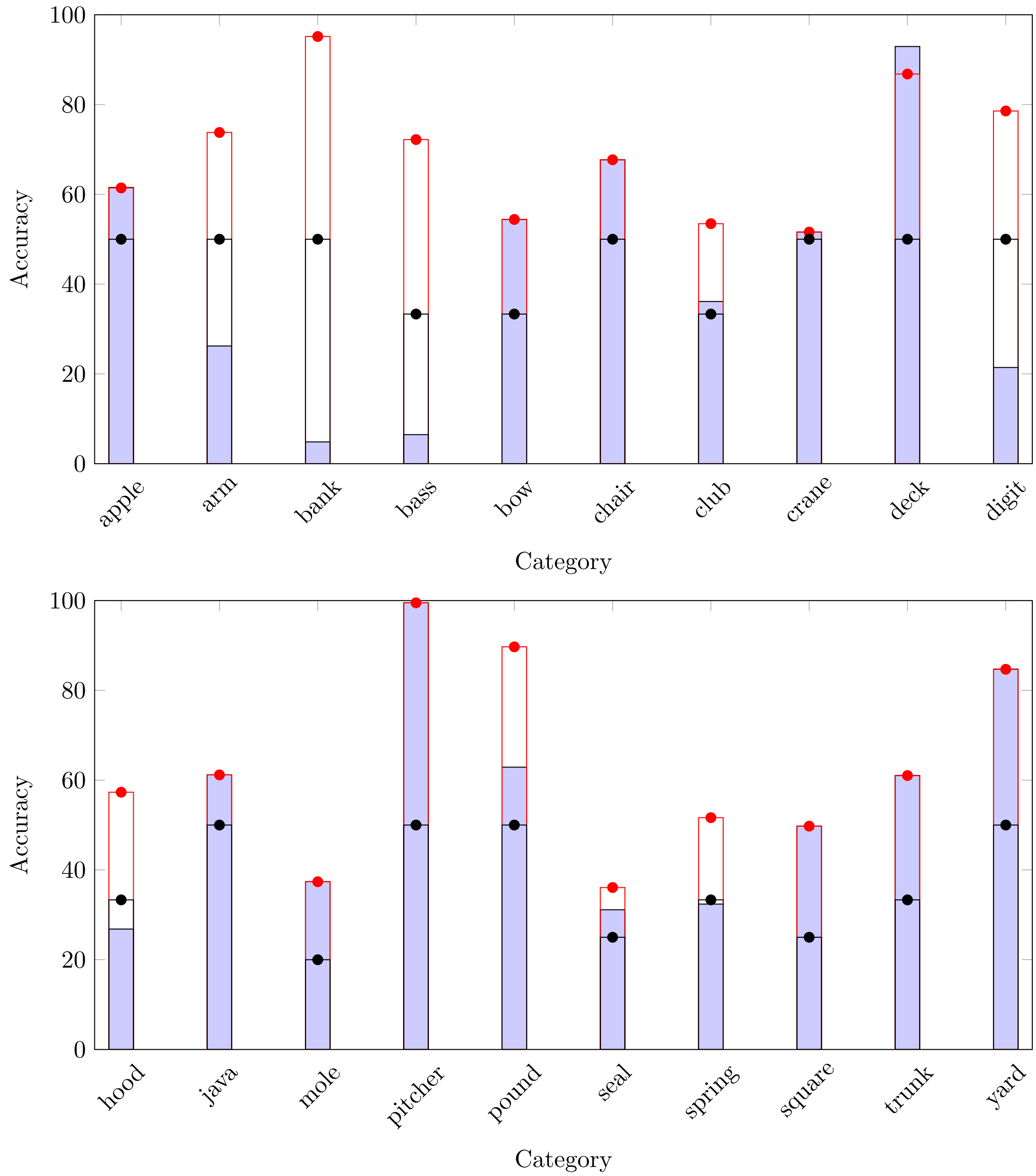

Figure 4: Results obtained for the CoarseWSD-20 dataset using UWSD+WMD. The solid blue bar represents the results obtained. While the black and red colors represent the weak and strong baselines, respectively

Strategy	Hits	Accuracy
UWSD+BERT+all-mpnet-base-v2	7,927	77.74%
UWSD+BERT+all-MiniLM-L12-v2	7,652	75.05%
UWSD+BERT+all-MiniLM-L6-v2	7,609	74.63%
MFS-Baseline	7,487	73.43%
UWSD+BERT+paraphrase-albert-small-v2	7,104	69.67%
UWSD+BERT+paraphrase-MiniLM-L3-v2	7,098	69.62%
UWSD+BERT+all-distilroberta-v1	5,547	54.40%
RO-Baseline	4,459	43.73%

Table 2: Summary of the results global results obtained using different language models based on BERT

6. Discussion

Our strategy has exhibited promising results in enhancing the accuracy of WSD. By considering the interplay of words within a specific context, the proposed method can effectively capture the subtle distinctions in word senses, thereby leading to more precise disambiguation.

One advantage of the proposed strategy is its ability to function without annotated data, making it more widely applicable to diverse languages and domains. This is particularly crucial for low-resource languages where annotated data is scarce and expensive to obtain.

Another crucial advantage of our method is its capability to capture the intricate nuances of meaning that traditional semantic models might overlook. By incorporating contextual information, we can differentiate between polysemous words with various meanings in different contexts. This is particularly significant in applications where accurate representation of meaning is crucial.

The experimental results demonstrate that the proposed method outperforms several unsupervised WSD techniques on the CoarseWSD-20 benchmark dataset. The performance improvement is particularly remarkable for words with high levels of ambiguity, where conventional semantic similarity methods often struggle to disambiguate accurately.

While the proposed approach displays promise, some limitations still require addressing. One limitation is that the approach heavily relies on the quality of available contextual information. Disambiguation accuracy may be compromised when the context is noisy or ambiguous. Additionally, the proposed approach may perform inadequately in cases where the context is too sparse or specific, resulting in inadequate information for precise disambiguation.

Future research directions could explore further enhancements to the approach, such as integrating additional sources of contextual information or combining it with other WSD techniques. Our method can significantly enhance the performance of many NLP tasks. Therefore, we plan to investigate its applicability to other domains and explore ways to improve its performance. Another challenge is the scalability of the method as the volume of text data continues to expand.

7. Conclusion

This work demonstrates that CASS enables machines to process and comprehend human language more effectively since it can recognize that the meaning of a word can differ based on the specific context in which it appears. CASS techniques aim to capture this variability, produce more precise similarity scores, and enhance the performance of unsupervised WSD strategies.

The proposed method achieved significant improvements in disambiguation accuracy compared to conventional methods that do not consider contextual information. The fact that the proposed method performed better than numerous other unsupervised strategies is further evidence of the method's usefulness in dealing with the ambiguity of word senses.

We have seen that including contextual information is a proven way to enhance accuracy when facing unsupervised WSD tasks. Our research suggests that using a strategy of this kind can address the challenge of interpreting the meaning of language as used in particular settings. The ability of computers to do so enables them to give users results that are more accurate and relevant, increasing the effectiveness and efficiency of various applications.

To conclude, CASS and WSD have a great deal of untapped potential that may improve the accuracy of text understanding and inspire the development of novel NLP applications. Novel strategies in this direction could lead to the development of solutions that are smarter, more efficient, and better able to comprehend the complexities and nuances of human language.

Source code

The source code of this approach is published under MIT license in the following GitHub repository: `https://github.com/jorge-martinez-gil/uwsd`.

Acknowledgments

The research reported in this paper has been funded by the Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation, and Technology (BMK), the Federal Ministry for Digital and Economic Affairs (BMDW), and the State of Upper Austria in the frame of SCCH, a center in the COMET - Competence Centers for Excellent Technologies Programme managed by Austrian Research Promotion Agency FFG.

References

[1] Apidianaki, M. (2022). From word types to tokens and back: A survey of approaches to word meaning representation and interpretation. *Computational Linguistics*, (pp. 1–60).

[2] Bollegala, D., Matsuo, Y., & Ishizuka, M. (2011). A web search engine-based approach to measure semantic similarity between words. *IEEE Trans. Knowl. Data Eng.*, *23*, 977–990. doi:10.1109/TKDE.2010.172.

[3] Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R. S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Strope, B., & Kurzweil, R. (2018). Universal sentence encoder for english. In E. Blanco, & W. Lu (Eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018* (pp. 169–174). Association for Computational Linguistics. doi:10.18653/v1/d18-2029.

[4] Chandrasekaran, D., & Mago, V. (2021). Evolution of semantic similarity - A survey. *ACM Comput. Surv.*, *54*, 41:1–41:37. doi:10.1145/3440755.

[5] Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)* (pp. 4171–4186). Association for Computational Linguistics. doi:10.18653/v1/n19-1423.

[6] Eyal, M., Sadde, S., Taub-Tabib, H., & Goldberg, Y. (2021). Large scale substitution-based word sense induction. *arXiv preprint arXiv:2110.07681*, .

[7] Han, L., Kashyap, A. L., Finin, T., Mayfield, J., & Weese, J. (2013). Umbc_ebiquity-core: Semantic textual similarity systems. In M. T. Diab, T. Baldwin, & M. Baroni (Eds.), *Proceedings of the Second Joint Conference on Lexical and Computational Semantics, *SEM 2013, June 13-14, 2013, Atlanta, Georgia, USA* (pp. 44–52). Association for Computational Linguistics.

[8] Han, S., & Shirai, K. (2021). Unsupervised word sense disambiguation based on word embedding and collocation. In A. P. Rocha, L. Steels, & H. J. van den Herik (Eds.), *Proceedings of the 13th International Conference on Agents and Artificial Intelligence, ICAART 2021, Volume 2, Online Streaming, February 4-6, 2021* (pp. 1218–1225). SCITEPRESS. URL: https://doi.org/10.5220/0010380112181225. doi:10.5220/0010380112181225.

[9] Harispe, S., Ranwez, S., Janaqi, S., & Montmain, J. (2015). *Semantic Similarity from Natural Language and Ontology Analysis*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers. doi:10.2200/S00639ED1V01Y201504HLT027.

[10] Kusner, M., Sun, Y., Kolkin, N., & Weinberger, K. (2015). From word embeddings to document distances. In *International conference on machine learning* (pp. 957–966). PMLR.

[11] Lastra-Díaz, J. J., & García-Serrano, A. (2015). A new family of information content models with an experimental survey on wordnet. *Knowl.-Based Syst.*, *89*, 509–526. doi:10.1016/j.knosys.2015.08.019.

[12] Lastra-Díaz, J. J., García-Serrano, A., Batet, M., Fernández, M., & Chirigati, F. (2017). HESML: A scalable ontology-based semantic similarity measures library with a set of reproducible experiments and a replication dataset. *Inf. Syst.*, *66*, 97–118. doi:10.1016/j.is.2017.02.002.

[13] Lastra-Díaz, J. J., Goikoetxea, J., Taieb, M. A. H., García-Serrano, A., Aouicha, M. B., & Agirre, E. (2019). A reproducible survey on word embeddings and ontology-based methods for word similarity: Linear combinations outperform the state of the art. *Eng. Appl. Artif. Intell.*, *85*, 645–665. doi:10.1016/j.engappai.2019.07.010.

[14] Loureiro, D., Jorge, A. M., & Camacho-Collados, J. (2022). Lmms reloaded: Transformer-based sense embeddings for disambiguation and beyond. *Artificial Intelligence*, *305*, 103661.

[15] Loureiro, D., Rezaee, K., Pilehvar, M. T., & Camacho-Collados, J. (2021). Analysis and evaluation of language models for word sense disambiguation. *Computational Linguistics*, *47*, 387–443.

[16] Martinez-Gil, J. (2022). A comprehensive review of stacking methods for semantic similarity measurement. *Machine Learning with Applications*, *10*, 100423.

[17] Martinez-Gil, J., & Chaves-Gonzalez, J. M. (2020). A novel method based on symbolic regression for interpretable semantic similarity measurement. *Expert Syst. Appl.*, *160*, 113663. doi:10.1016/j.eswa.2020.113663.

[18] Martinez-Gil, J., & Chaves-Gonzalez, J. M. (2021). Semantic similarity controllers: On the trade-off between accuracy and interpretability. *Knowl. Based Syst.*, *234*, 107609. doi:10.1016/j.knosys.2021.107609.

[19] Martinez-Gil, J., & Chaves-Gonzalez, J. M. (2022). Sustainable semantic similarity assessment. *Journal of Intelligent & Fuzzy Systems*, *43*, 6163–6174. doi:10.3233/JIFS-220137.

[20] Martinez-Gil, J., Mokadem, R., Küng, J., & Hameurlain, A. (2021). A novel neurofuzzy approach for semantic similarity measurement. In M. Golfarelli, R. Wrembel, G. Kotsis, A. M. Tjoa, & I. Khalil (Eds.), *Big Data Analytics and Knowledge Discovery - 23rd International Conference, DaWaK 2021, Virtual Event, September 27-30, 2021, Proceedings* (pp. 192–203). Springer volume 12925 of *Lecture Notes in Computer Science*. doi:10.1007/978-3-030-86534-4_18.

[21] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.* (pp. 3111–3119).

[22] Moradi, B., Ansari, E., & Zabokrtský, Z. (2019). Unsupervised word sense disambiguation using word embeddings. In *25th Conference of Open Innovations Association, FRUCT 2019, Helsinki, Finland, November 5-8, 2019* (pp. 228–233). IEEE. URL: https://doi.org/10.23919/FRUCT48121.2019.8981526. doi:10.23919/FRUCT48121.2019.8981526.

[23] Navigli, R. (2009). Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, *41*, 1–69.

[24] Navigli, R., & Martelli, F. (2019). An overview of word and sense similarity. *Nat. Lang. Eng.*, *25*, 693–714. doi:10.1017/S1351324919000305.

[25] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. In M. A. Walker, H. Ji, & A. Stent (Eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)* (pp. 2227–2237). Association for Computational Linguistics. doi:10.18653/v1/n18-1202.

[26] Pilehvar, M. T., & Navigli, R. (2015). From senses to texts: An all-in-one graph-based approach for measuring semantic similarity. *Artif. Intell.*, *228*, 95–128. doi:10.1016/j.artint.2015.07.005.

[27] Rahman, N., & Borah, B. (2022). An unsupervised method for word sense disambiguation. *J. King Saud Univ. Comput. Inf. Sci.*, *34*, 6643–6651. URL: https://doi.org/10.1016/j.jksuci.2021.07.022. doi:10.1016/j.jksuci.2021.07.022.

[28] Skianis, K., Malliaros, F. D., Tziortziotis, N., & Vazirgiannis, M. (2020). Boosting tricks for word mover's distance. In *International Conference on Artificial Neural Networks* (pp. 761–772). Springer.

[29] Ustalov, D., Teslenko, D., Panchenko, A., Chernoskutov, M., Biemann, C., & Ponzetto, S. P. (2018). An unsupervised word sense disambiguation system for under-resourced languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA). URL: http://www.lrec-conf.org/proceedings/lrec2018/summaries/182.html.

[30] Zhu, G., & Iglesias, C. A. (2017). Computing semantic similarity of concepts in knowledge graphs. *IEEE Trans. Knowl. Data Eng.*, *29*, 72–85. doi:10.1109/TKDE.2016.2610428.