Journal of King Saud University - Computer and Information Sciences 36 (2024) 101911

HOSTED BY

Contents lists available at ScienceDirect

Journal of King Saud University - Computer and Information Sciences

journal homepage: www.sciencedirect.com

Full Length Article

EnhancedBERT: A feature-rich ensemble model for Arabic word sense disambiguation with statistical analysis and optimized data collection

Sanaa Kaddoura *, Reem Nassar

Zayed University, Abu Dhabi, United Arab Emirates

ARTICLE INFO

Keywords:
Arabic natural language processing
Word sense disambiguation
Machine learning
Knowledge-based
BERT
Performance evaluation

ABSTRACT

Accurate assignment of meaning to a word based on its context, known as Word Sense Disambiguation (WSD), remains challenging across languages. Extensive research aims to develop automated methods for determining word senses in different contexts. However, the literature lacks the presence of datasets generated for the Arabic language WSD. This paper presents a dataset comprising a hundred polysemous Arabic words. Each word in the dataset encompasses 3–8 distinct senses, with ten example sentences per sense. Some statistical operations are conducted to gain insights into the dataset, enlightening its characteristics and properties. Subsequently, a novel WSD approach is proposed to utilize similarity measures and find the overlap between contextual information and dictionary definitions. The proposed method uses the power of BERT, a pre-trained language model, to enable effective Arabic word disambiguation. In training, new features are integrated to improve the model's ability to differentiate between various senses of words. The proposed BERT models are combined to compose an ensemble model architecture to improve the classification performances. The performance of the WSD system outperforms state-of-the-art systems, achieving an approximate F1-score of 96 %. Statistical analyses are performed to evaluate the overall performance of the WSD approach by providing additional information on model predictions. A case study was implemented to test the effectiveness of WSD in sentiment analysis, a downstream task.

1. Introduction

Polysemy, the phenomenon of words having multiple meanings, is prevalent in all languages Kaddoura et al. (2022). It allows for detailed expressions and various interpretations to enhance the language. In English, more than 40 % of words are polysemous Abou Khalil et al. (2019). Arabic, a language renowned for its complexity, has a higher proportion of polysemantic words, manifesting ambiguity at various levels Farghaly et al. (2009). Different forms of ambiguity, e.g., homograph, internal word structure, syntactic, semantic, constituent boundary, and anaphoric ambiguity, are shown in Arabic Farghaly et al. (2009). The average degree of ambiguity in Modern Standard Arabic (MSA) tokens is 19.2 % ambiguities, far exceeding 2.3 % in most languages Farghaly et al. (2009).

While humans possess an inborn ability to discern the intended sense of polysemous words, computers face challenges in resolving such ambiguities. Computers treat words with the same letters as a single entity, ignoring context, unlike humans, who unconsciously deal with ambiguity during language acquisition. Despite significant research efforts, this discrepancy underscores the ongoing difficulty of Arabic Word Sense Disambiguation (WSD) Debili et al. (2002). The Arabic language utilizes diacritics to aid pronunciation and comprehension. For example, consider the word "هبة" (hibap) as an example of ambiguity. It poses a different meaning when adding a diacritic mark () above the second letter and is thus written and pronounced as "هبة" (hab ~ ap). Nevertheless, computers are hampered by the absence of diacritics in various published texts, including news, scientific articles, comics, and many other Arabic texts. Studies have shown that 43 % of diacretized Arabic words exhibit ambiguity, while it rises to 72 % when diacritic marks are missing, which is common in written Arabic text Alqahtani et al. (2019).

Consider the following examples illustrating the different meanings of the word "هبة" (hibap) without using diacritics. In the sentence "ساهمت المهندسة هبة في الأنشطة البيئية في المدرسة أو المخيمات الصيفية" (Engineer Hiba contributed to environmental activities in school or summer camps), "هبة" (hibap) refers to a proper name. However, in "قدم الملك هبة قيمتها مليون دولار للدول النامية" (The king provided a donation worth a million dollars to developing countries), "هبة" (hibap) signifies giving. Moreover, in "دعا الحزب كافة الشعب إلى هبة شعبية تنضم إلى المظلومين"

* Corresponding author.
E-mail address: Sanaa.kaddoura@zu.ac.ae (S. Kaddoura).

https://doi.org/10.1016/j.jksuci.2023.101911
Received 14 October 2023; Received in revised form 21 December 2023; Accepted 30 December 2023
Available online 8 January 2024

(The party called upon the entire nation for a popular strike in solidarity with the oppressed), "هبة" (hab ~ ap) implies a strike. One can successfully predict the intended sense by considering the sentence content words.

In many applications, including machine translation, text summary, information retrieval, and query answering, the functionality of WSD Systems in Computational Linguistics has a critical role Ide and Véronis (1998); AlMousa et al. (2022). While the Large Language Models (LLM) can accomplish these tasks without WSD, integrating WSD models enhances performance. WSD approaches offer several advantages to LLMs in downstream tasks, including improving model accuracy, contextual comprehension, and addressing polysemy. For example, consider the sentence "رأيت الأب وهو خارج من الكنيسة," where the polysemous word "الأب" can refer to "father" or "church father/priest." In machine translation using LLMs, WSD prevents inaccuracies, ensuring the correct rendering of the sentence as "I saw my father coming out of the church" rather than "I saw the priest coming out of the church." This misinterpretation results from a misunderstanding of the polysemous word Kaddoura et al. (2023). Furthermore, LLM hallucinations can take various forms, including generating factually incorrect text, inventing unrelated stories, and even generating non-existent images.

The inherent complexity and scarce resources have made it difficult to advance research into WSD for the Arabic language Djaidri et al. (2023). Therefore, Arabic Natural Language Processing (NLP) often experiences challenges in delivering expected results. Currently, there is a lack of an efficient WSD system tailored explicitly for the Arabic language El-Razzaz et al. (2021), consequently limiting the effectiveness of tasks like information retrieval. There is still a significant lack of a comprehensive lexical database that organizes Arabic words into a set of synonyms based on common meanings Boudabous et al. (2013), despite previous attempts to explore WSD in Arabic. The absence of a word sense corresponding to polysemous Arabic words makes it more difficult to detect them accurately. Therefore, a research gap exists in the domain of WSD in Arabic, and that is due to a need for more resources.

WSD approaches are classified into four categories: supervised Yarowsky (1995), unsupervised Pedersen (2007), semi-supervised Taghipour and Ng (2015), and knowledge-based Banerjee and Pedersen (2003). Hybrid WSD systems are also implemented by combining two WSD categories. Knowledge-based methods use language resources such as knowledge graphs and dictionaries. The knowledge-based approach defines two categories: graph-based subcategories and sense-based subcategories. Graph-based techniques use language knowledge graphs such as WordNet to represent senses derived from their interaction in the graph. WordNet lacks entries for less common or domain-specific terms, making it difficult to disambiguate word senses accurately. Sense-based approaches assess semantic similarity and overlap between contexts of a given ambiguous word. These methods may also not fully consider important syntactic and pragmatic context cues vital for accurate sense disambiguation. Unsupervised WSD struggles in highly ambiguous contexts, often offering multiple unranked candidate senses. It can also perform poorly in domain-specific contexts and may miss subtle contextual cues essential for accurate sense disambiguation. Semi-supervised approaches are complex to implement and fine-tune compared to supervised methods. When trained on quality labeled data, supervised WSD methods are accurate and precise, making them ideal for clear contexts. They are adaptable to specific domains or languages, handle diverse word senses, including fine-grained distinctions, and can effectively generalize to unseen words using contextual cues. Obtaining contextual cues in supervised learning can be challenging, depending on the specific task and the available data.

This paper addresses the challenge of a scarcity of Arabic WSD corpora, the necessity for a substantial corpus, and the critical importance of understanding contextual cues. The contributions are as follows:

1. Introduce an extensive dataset featuring one hundred commonly used polysemantic Arabic words in MSA. Each word in the dataset is meticulously annotated with three to eight senses.
2. Employ BERT as the foundational model to disambiguate polysemous words.
3. Incorporate diverse data representation techniques and introduce new feature sets that enhance the disambiguation process by providing valuable contextual information.
4. Propose an ensemble learning architecture that combines multiple BERT models to address word ambiguity effectively.
5. Evaluate WSD models by comparing the proposed approach with the other benchmark WSD systems.
6. Conduct statistical analyses as a post-processing step to evaluate the performance of the proposed WSD approach and gain insights into its effectiveness in disambiguating word senses.
7. Conduct a case study to see the effectiveness of WSD in improving downstream tasks like sentiment analysis.

Ten instances are extracted for each target word in the collected dataset to ensure a robust analysis. Incorporating various features and data representation techniques empowers the model to assign increased attention weights to context and ambiguous words.

The research questions in this paper are as follows:

RQ1: How does the newly created dataset of frequently occurring Arabic words in MSA contribute to the Arabic WSD field?
RQ2: How do different data representation techniques impact the suitability of the dataset for Arabic WSD?
RQ3: How does applying BERT models enhance the disambiguation process?
RQ4: To what extent does the inclusion of part-of-speech (POS) information improve the accuracy and effectiveness of Arabic WSD models based on BERT?
RQ5: How does considering word frequency affect the performance of Arabic WSD models, mainly when dealing with rare words?
RQ6: What benefits does the weighted ensemble approach offer in Arabic WSD, and how does it contribute to the disambiguation of specific word senses?
RQ7: Under what conditions and to what extent do BERT-based approaches outperform traditional methods, like Naive Bayes, in Arabic WSD?
RQ8: What are the main errors of the proposed Arabic WSD system, and do they reveal common patterns or linguistic challenges that need to be addressed?
RQ9: How does the scarcity of Arabic language resources, such as sense-annotated corpora and lexicons, affect the performance and feasibility of Arabic WSD systems?

The rest of the paper is organized as follows: Section 2 presents the literature for Arabic WSD datasets and a review of the existing Arabic WSD approaches, Section 3 describes the Arabic morphology, Section 4 reports the dataset collection criteria and the statistical analyses for the collected dataset, Section 5 explains the proposed WSD methodology, Section 6 displays the experimental results, Section 7 presents the case study implemented, Section 8 provides the discussion for WSD, and Section 9 presents the conclusion.

2. Related work

2.1. Review of existing datasets

External knowledge sources are classified into four categories under WSD: structured, unstructured, WordNet, and Semcor. Structured resources provide word-related information that facilitates word clustering based on semantic similarity, including thesauri, machine dictionaries, and ontologies Kilgarriff and Yallop (2000). The valuable

sources of textual content for research on WSD are unstructured resources, such as corpora and word frequency lists. WordNet, widely used in WSD, organizes concepts into synsets and offers semantic and lexical relations, aiding in sense disambiguation across various languages Gonzalo et al. (2000). SemCor, an enhanced version of WordNet, provides a semantically defined set of corpora for supervised WSD techniques, offering valuable training and evaluation resources Gonzalo et al. (2000).

Due to the scarcity of extensively annotated corpora, WSD in Arabic presents a critical challenge. Researchers have undertaken several efforts to create suitable datasets for WSD studies. Nevertheless, progress in this area has been limited because these data sources are frequently unavailable or unretrievable. Some studies used Arabic Wordnet Fellbaum et al. (2006) for sense annotations based on text classification data. However, the Arabic Wordnet Fellbaum et al. (2006) is not a comprehensive lexical database for grouping Arabic words with similar meanings into synonym sets, which makes it difficult to make accurate sense predictions Bouhriz et al. (2016).

Researchers have applied various approaches to collecting Arabic WSD data with the intention of dealing with the problem of limited data availability. Grave et al. (2018) developed a multilingual distributed word representation by collecting data from Wikipedia and the Common Crawl project. Fouad et al. (2020) proposed contextual word embeddings for Arabic by gathering data from Twitter. Habib et al. (2021) collected 1.5 million medical questions from the Altibbi Medical Company to create contextual embeddings. Alian et al. (2019) collected sentences from books used for teaching syntax and semantics in Arabic, initially designed for semantic similarity and paraphrasing tools but can also be used for WSD.

Moreover, Belinkov et al. (2016) developed a comprehensive Arabic corpus by collecting data from "The Complete Library Website." Abu El-Khair (2016) constructed another extensive Arabic corpus of 1.5 billion words through web scraping from diverse domains. However, some of these resources remain private, and the public ones require comprehensive human sense annotation due to their size and the absence of specific annotations for WSD.

Saidi et al. (2023) introduced a manually curated corpus designed for Arabic WSD. The polysemous words in the corpus are extracted from the Doha historical dictionary for Arabic. The proposed corpus comprises 7,721 Arabic polysemous words. Each word within the corpus is linked to various senses, accompanied by corresponding contextual examples. The corpus encompasses 27,530 total sentences and 16,316 total senses. The distribution of context examples across these senses is not uniform, leading to an average of 1.68 sentences per sense. This variance implies that a more significant number of contextual instances represents certain senses compared to others. This non-uniform distribution of sentences per sense will introduce a potential bias in machine learning models trained on this corpus. The model may inherently favor senses with a higher abundance of examples, as it tends to learn more from them during the training process.

An alternative approach for constructing sense-tagged corpora in WSD is the knowledge-based method. This method was introduced by Saif et al. (2018). Their method focuses on creating an Arabic sense-tagged corpus derived from Wikipedia. This approach hinges on Arabic WordNet, where a mapping between WordNet and Wikipedia articles is established to select the appropriate sense for each article based on WordNet. This mapping leverages a cross-lingual method to measure the similarity between the features within the Wikipedia articles and WordNet senses. A multiword-based technique is proposed to address the challenge of limited instances in Wikipedia articles. This technique is a valuable resource, increasing the number of example instances for each sense by considering multiword terms. The resultant sense-tagged corpus encompasses 50 Arabic words, containing 148 senses and a remarkable 30,961 samples. Although the corpus is substantial, it does not comprehensively cover all the senses for each word. On average, the proposed data contains 2.9 senses per word, while many Arabic words exhibit more than three senses.

2.2. Review of word disambiguation techniques

Early research since the mid-1940s has introduced the cruciality of addressing word ambiguity across multiple languages. Zipf (1945) proves the relation between word frequency and the number of meanings. Zipf's introduced the principle of meaning distribution, where words that occur more frequently typically have more meanings. English consistently leads in WSD research compared to other languages Miller (1992). However, progress in research on WSD continues to be limited, although significant efforts have been made in languages such as Arabic. Three distinct main categories of approaches to WSD can be classified: (1) Machine Learning, (2) knowledge-based methodology, and (3) hybrid methods.

2.2.1. Machine learning approaches

Different machine learning techniques for WSD are used, including unsupervised, supervised, and semi-supervised methods. Supervised methods offer higher accuracy despite the need for a comprehensive sense annotated corpus. El-Gedawy (2013) proposed a supervised approach that utilized fuzzy logic classifiers and relied on an English WordNet-based sense inventory to assign accurate senses to polysemous words, achieving an F1-score of 74 % in their evaluation. Another supervised WSD algorithm based on the Naïve Bayes Classifier was employed by Elmougy, Taher, and Noaman (2008). The proposed method yields a precision of 73 % with an improvement when combined with the rooting algorithm. To address morphosyntactic disambiguation, Albared et al. (2009) studied the integration of multiple classifiers, including maximum entropy hidden Markov and transformation-based probabilistic classification, which implements preprocessing steps like voting tags and cascading along with a selection algorithm. Their study addressed data sparseness by using different hidden Markov models and selecting the most likely POS tag based on contextual information in the text. It achieved an accuracy of approximately 95.8 %. Saidi, Jarray, and Alsuhaibani (2022) addressed this problem using four Recurrent Neural Network (RNN) architectures, including Vanilla RNN, LSTM, BiLSTM, and GRU, for supervised sequence-to-sequence learning. The GRU-based deep learning model demonstrates superiority over existing RNN models by achieving an accuracy rate of 92.83 %, which is higher by 2.06 %, 4.29 %, and 7.61 % than BiLSTM, LSTM, and Vanilla RNN, respectively.

Unsupervised approaches are used in WSD to circumvent the requirement of annotating senses from a dataset. An early unsupervised approach for Arabic WSD was suggested by Diab and Resnik (2002). The proposed system established a strong correlation between the meaning dimension of words and their translations. Their method operated under the assumption that words with similar translations share parallel dimensions of meaning through leveraging English WordNet for word senses and an English-Arabic corpus for translation. Alian and Awajan (2023) have implemented a disambiguation method using an unsupervised methodology for synthesizing the sense inventory based on predefined embedded patterns. To enhance the selection of appropriate senses, they used POS tags for identified senses and compared them to those indicating an ambiguous word. As shown by Pearson Correlation, experimental tests focused on the similarity of sentences when the selected sense vector was superior to using an ambiguous word vector in significantly improving sentence similarity. The use of Aravec-embedded modules reached an increased correlation value of 0.423. Al-Maghasbeh and Bin-Hamzah (2015) suggested a novel method that extracts prepositions to analyze Arabic texts and establishes the relationship between prepositions and sentence components, facilitating the identification of correct senses for the prepositions. In addition to the dictionary of prepositions, this methodology was tested in samples from the Holy Quran containing words where each has several meanings.

Semi-supervised approaches have been proposed in the WSD due to data scarcity. A technique based on word embeddings for Arabic WSD

has been developed by Laatar et al. (2018). This approach aimed to create a historical dictionary with corresponding meanings for ambiguous Arabic words. The word embeddings were obtained by capturing the word's semantic and syntactic characteristics and thus achieving an accuracy of 78 %. Merhbene et al. (2014) developed an Arabic WSD using a semi-supervised approach incorporating a dictionary of words and a corpus for classification. This approach also includes semantic trees for representing each sense. Merhbene et al. (2013) introduced a weighted directed graph-based method using the Arabic WordNet Fellbaum et al. (2006) and a corpus. The Arabic corpus is utilized to build clusters of senses that correspond to some ambiguous Arabic word senses to describe. A recall score of 83 % was achieved for classification, higher than that reported by k-nearest neighbor algorithms.

All the previously mentioned machine learning approaches have based their model assessment on performance metrics like accuracy and F1-score. However, in the case of WSD, one must rely on more than just high values for model evaluation. Arabic polysemous words have a minimum of two senses per word. This difference between senses will lead to unbalanced WSD datasets where some polysemous words have two senses and others more. Consider a dataset where word senses range from 2 to 8. The inability to disambiguate a few words with only two senses has a relatively minor impact on overall model evaluation metrics due to their lower weight than the rest of the dataset. Therefore, a comprehensive assessment of Arabic WSD models must be considered.

2.2.2. *Knowledge-based approaches*

In Arabic, various WSD knowledge-based approaches have been proposed. Alian et al. (2016) developed a method that utilizes open-source resources and retrieves ambiguous word senses from Wikipedia. The text is preprocessed, and senses are described as vectors, with a cosine similarity applied to disambiguation. Their algorithm has three steps: text preprocessing and ambiguous word determination, searching for the ambiguous word in Wikipedia, and applying vector space model and similarity measures to extract the meaning. Alkhatlan et al. (2018) incorporated state-of-the-art techniques such as GloVe. They addressed the limitation of representing word meanings as a single vector by leveraging Arabic WordNet Fellbaum et al. (2006) to compute embeddings for each sense.

Zouaghi et al. (2011) introduced a gloss-based approach to computing the overlap of sense definitions, using resources such as the AWN and the Simlch score. They assessed the variants of the LESK algorithm by experimenting and introducing modifications to disambiguate Arabic words. The original LESK algorithm achieved a 59 % accuracy by utilizing an Arabic dictionary and Al-Sulaiti and Atwell corpus Al-sulaiti and Atwell (2003). Abdelaali and Tlili-Guiassa (2022) used gloss and context overlap and proposed a method based on the LESK and Cuckoo Optimization Algorithm. English pre-trained word embeddings were used due to a shortage of Arabic lexical resources. Their results show a significant improvement compared with the baseline method. These knowledge-based methods can provide valuable information and contribute to the Arabic WSD field. However, these approaches rely on lexical resources, and the Arabic language still lacks the presence of a comprehensive resource. The available lexical resources only include some Arabic words or some senses per word. Additionally, relying on English lexical resources might not be efficient since machine translation depends on WSD and can yield incorrect results. On the other hand, manually translating Arabic text requires human intervention, effort, and time.

Graph-based approaches have also been proposed in the literature. Corrêa Jr., Lopes, and Amancio (2018) introduced a statistical model to address textual ambiguities by focusing on the semantic relationships between feature words and target words. This method employs graph networks for representing text and establishes a structure for disambiguating word senses. The graph network comprises layers representing feature words and target words, emphasizing their semantic connections while overlooking relationships among feature words. Notably, their approach demonstrates promising performance, especially when utilizing local word extraction to capture semantic contexts. This method proved effective in disambiguating words, even when faced with limited data, and in specific cases, it outperformed the support vector machine-based WSD model.

Corrêa Jr. and Amancio (2019) proposed a disambiguation method that uses recent findings in word embeddings to create context embeddings. Their disambiguation algorithm does not rely on any structured knowledge sources. They employed a bipartite network framework to tackle word sense disambiguation, representing ambiguous words as nodes connected when their context embeddings demonstrate similarity. Subsequently, they applied a clustering algorithm to these embeddings to identify the precise sense for each ambiguous term. The proposed method outperformed other state-of-the-art algorithms.

A graph-based WSD method with multi-knowledge integration is proposed by Lu et al. (2019). The proposed graph model combines diverse Chinese and English knowledge resources through word sense mapping. Three evaluative word similarity measures characterize their method. First, the content words in an ambiguous Chinese sentence are identified and linked to their English counterparts using BabelNet. Then, English word similarity is calculated based on English word embeddings and the knowledge base. Chinese word similarity is computed using Chinese word embeddings and HowNet. The word similarity weights are optimized through a simulated annealing algorithm to derive their overall similarities, which are employed in the creation of a disambiguation graph. This graph is then evaluated by a graph scoring algorithm, which determines the significance of each word sense node and identifies the correct senses for the ambiguous words.

Quispe, Tohalino, and Amancio (2021) proposed a method to enhance word co-occurrence networks for text analysis, offering valuable improvements for WSD. Enhancing this network is crucial for understanding semantic similarity among words. The methodology comprises four key steps: first, network construction, involving the mapping of texts into co-occurrence networks while considering or disregarding stop words; next, network enrichment, where virtual edges are introduced based on the similarity of word embeddings; followed by network filtering to eliminate spurious links; and concluding with feature extraction from the resulting network for pattern classification. The authors employed diverse word embeddings, such as GloVe, FastText, and Word2Vec, to measure semantic similarity using the cosine similarity measure between word vector representations. Their findings indicate that this approach significantly improved classification systems, providing compelling evidence for its efficacy in disambiguating words within a classification context.

2.2.3. *Hybrid approaches*

In the literature, a combination of different WSD methods, known as hybrid approaches, has been proposed. A hybrid method combining knowledge-based and unsupervised techniques to develop an Arabic WSD system was presented by Zouaghi, Merhbene, and Zrigui (2012a). The unsupervised method focuses on word sense recognition possibilities, while the knowledge-based approach selects the most suitable sense based on options given by an undirected algorithm. A 73 % accuracy was achieved with the hybridization of LESK. Another hybrid algorithm combining knowledge methods, such as conceptual density and random walk, with graph methods has been introduced by Abderrahim and Mohammed El Amine (2022). They have utilized the Arabic Wordnet Fellbaum et al. (2006) to get the word senses. Their WSD algorithm increased their information retrieval system by 12 % in F1-score terms based on a medium-sized corpus. Zouaghi, Merhbene, and Zrigui (2012b) proposed a hybrid approach combining knowledge-based and unsupervised WSD methods. They applied preprocessing steps for ambiguous corpus texts, identified critical words that affected their meaning, and used a context-matching algorithm. This algorithm will calculate a semantic coherence score to determine the context of use that is semantically equivalent to the first words. The system has

achieved a precision of 79 %. These hybrid approaches suffer from complexity as they combine two different WSD approaches.

3. Arabic morphology

3.1. Sense variation

The Arabic language is rich and complex, carrying out different senses of polysemous words at multiple levels. According to Farghaly et al. (2009), senses may vary based on multiple conditions. The sense variation is based on the following:

Interdomain information ambiguity: Arabic words carry different meanings in various domains or fields of knowledge. For example, the Arabic word "مفتاح" (miftAH) has multiple senses that can refer to a physical object (Key) or a domain of information retrieval or computer programming (Keyword).
POS tag ambiguity: Arabic words have different meanings based on their grammatical function in a sentence. For example, the term "ذهب" (*ahab) can be a verb referring to an act of leaving or a noun that means gold.
Named entity ambiguity: a single Arabic word might refer to multiple entities, such as people, animals, places, and organizations. For example, the name "رشا" (ra$A) can refer to a person's name or an animal name, the small dear.
Content and function word ambiguity: Arabic words can function as content words (nouns, verbs, adjectives, and adverbs) and function words (pronouns, prepositions, conjunctions, articles, and auxiliary verbs). For example, "ألم" (>lam) can be a content word as a noun referring to pain, an interrogative pronoun used to ask for information or a relative pronoun that introduces a clause that describes or provides additional information.

3.2. Challenges in Arabic

A word in Arabic language is complex, and its complexity affects the performance of WSD systems. This difficulty has made the performance of Arabic NLP systems lag behind the literature in other languages. The language complexity occurs due to its agglutinative nature, inflectional properties, and the removal of diacritics that aid the pronunciation. Normalizing Arabic text in most published Arabic articles has increased the language complexity. These language challenges introduce significant ambiguity, posing challenges for computers in predicting word senses accurately within the context. Table 1 provides examples from the web illustrating these three limitations in Arabic writing and how they affect ambiguity. Diacritics, which act as short vowels, are essential for conveying pronunciation, grammatical features, and word meanings in Arabic. Their absence in many text sources results in words that share identical spellings but possess different meanings and pronunciations, making it hard for machines to extract the intended sense. Table 1 provides an undiacretized example of the word "أجل" with three different meanings and pronunciations.

The normalization process is employed in Arabic text processing because Arabic letters can have different variants, such as "إ" (alif with hamza below), "أ" (alif with hamza above), "آ" (alif with madda), and other similar variations. These variants are transformed into the standard Arabic letter "ا" (alif) during normalization to ensure a consistent and uniform representation of words. Text in Arabic articles is often normalized to improve search and retrieval. This process often leads to the loss of sense information, further increasing ambiguity. One example of normalization is presented in Table 1. This example shows the removal of the hamza in the word "ثأر" (va > or), transforming it into "ثار" (vAra). Thus, obscuring the semantic and morphological distinctions between the two distinct words "ثأر" (va > or) and "ثار" (vAra); such normalization can pose challenges for WSD systems, as they may struggle to differentiate between words that have lost some of their original semantic structure.

Arabic's agglutinative nature, where prefixes, suffixes, and morphemes attach to root words to convey various meanings, generates many closely related word forms with distinct senses. This word representation increases the complexity of sense disambiguation. Table 1 shows two examples, first showing how suffixes and prefixes are added to the word as in "ويتذكرومم" (wayata*ak ~ arwhum) and "فأسقيناكموها" (fa > soqayonAkumwhA) that are variation of the base form "تذكر" (ta*ak ~ ara) and "سقى" (saqaY) respectively.

Arabic is a highly inflected language, which can introduce polysemy due to variations in number and gender. This inflection primarily occurs through the addition of suffixes, prefixes, or infixes to words. Consider the word "مدرسة" presented in Table 1; it can represent a female teacher, which is an inflected word derived from "مدرس," or it can refer to a school. This variation of senses highlights how even alterations in the structure of a word can result in entirely different interpretations or meanings. This characteristic of Arabic adds word ambiguity.

Table 1
Unique source of ambiguity in the Arabic language.

Ambiguity Source	Example	Polysemous word	Transliteration	Translation
Diacritics	أجل مجددا مؤتمر المصالحة الوطنية في الصومال.	أجل	>uj ~ ila	The national reconciliation conference in Somalia has been **postponed** again.
	أجل ما ينزل من السماء التوفيق.	أجل	>aj ~ al ~ a	Success is the most **sublime** thing that can be earned.
	لو لم يتم التسليم عند الأجل المتفق عليه بين البائع والمشتري فيحق للمشتري خيار فسخ العقد أو يأخذ الثمن.	أجل	>ajal	If delivery is not made by the **deadline** agreed upon between the seller and the buyer, the buyer can either cancel the contract or take the price back.
Normalization	وقد ثار المصري القديم لأسباب عديدة وأغراض متنوعة.	ثار	vAra	The ancient Egyptians **revolted** for many reasons and diverse purposes.
	ولعبت مؤسسة روبن مدريد الأمريكية دورا إيجابيا في إنعاش محاربة الثار عبر القبائل أنفسهم.	ثار	va > or	The Robin Madrid Foundation played a positive role in reviving the **fight** against revenge through the tribes.
Agglutinative	ويتذكرومم	تذكر	wayata*ak ~ arwhum	and they remember them.
	فأسقيناكموها.	سقى	fa > soqayonAkumwhA	So, we gave it to you to drink.
Inflection	تساهم المدرسة بشكل فعال في تنمية المجتمع، فهي المنبت الأساسي بعد الأسرة في تربية الأطفال، وتعليمهم، وصقل مواهبهم وشخصياتهم.	مدرسة	mdrsp	**School** plays a significant role in the development of society. It is the cornerstone, following the family, in raising and educating children, nurturing their talents, and shaping their personalities.
	إن غياب مدرسة مادة الرياضيات بعد فترة من بدء الفصل الدراسي قد أثر على أداء الطالبات في امتحان نهاية الفصل.	مدرسة	mdrsp	The absence of the mathematics **teacher** for a period after the start of the academic semester has affected the performance of the female students in the end-of-semester exam.

4. Dataset

4.1. Data collection

The Arabic language, considered a hot research area for the WSD, is challenging. Several studies have been conducted to resolve the problem of WSD, but there is a lack of publicly available data. Therefore, collecting a WSD dataset was an essential step toward developing accurate language models and algorithms that can be used effectively. For this purpose, a dataset of MSA's hundred most common polysemous words was collected. Each word has multiple senses or meanings based on the most commonly used senses in the language; the sense per word ranges between three and eight. Table 2 presents samples from the dataset.

Two methods are used for the creation of this dataset: Firstly, web scraping data and secondly, manual sorting of sentences and senses. The first step is to collect the polysemous words and their senses, which will form the basis of the dataset. A Python code that utilizes a library called Beautiful Soap is used to collect sentences from Wikipedia Foundation (2023) containing the target word. This approach leverages web scraping techniques to extract relevant sentences demonstrating the different senses of polysemous words. A substantial amount of data can be collected in this automatic approach. However, Wikipedia does not contain words with all their meanings, so manual sorting is used.

Following the data crawling step, manual sorting is performed to ensure the accuracy and relevance of the collected sentences. It involves carefully examining each sentence and assigning it to the proper meaning of the word based on electronic dictionaries like Almaany (2023) and Lexicon Alsharekh (2019). A native Arabic speaker collected the dataset. After that, another native Arabic expert volunteered to check the annotations for each polysemous word. In some cases, there may be missing sentences for certain senses. Therefore, further searches on the Internet are carried out to ascertain appropriate sentences representative of missing senses and complete the dataset. Data from various Web sites such as AlJazeera (2023b), Documentary AlJazeera (2023a), Altibbi (2023), Arabia Weather (2023), Shifa (2023), Arabia CNN (2023), Arabia BBC (2023), Arabic Post (2023), and Argaam (2023) are collected in this step. This step ensures that each polysemous word has comprehensive coverage of its senses. More than one website has been used to ensure that collected data is not limited to a single category or a particular way of writing.

Due to the limited availability of text on the Internet, several challenges have been encountered while collecting data. The dataset suffered from missing samples even after manual sorting, although the word senses are listed based on their popularity and usage in MSA. Therefore, the diversity and comprehensiveness of examples for certain senses will be restricted. Thus hindering the model's ability to learn and generalize accurately for those senses with limited examples. There are two limitations in this respect, for example:

Similar sentences from different websites were occasionally encountered, potentially introducing bias into the dataset. For instance, the word "أحد" (>uHud), when referring to the name of a mountain, was frequently paired with the term "battle" due to historical associations. This redundancy could mislead the model to associate the term ">uHud" solely with the concept of battle, leading to biased and inaccurate predictions.

Insufficient Sentence Count: Some words, such as "تاه" (tAha) with the meaning "Boastful," lacked the availability of 10 sentences in the dataset.

Efforts have been made to look for diverse Web data to alleviate these limitations. In case no data was found, GPT 3.5 was used to generate sentences where insufficient data exists. Algorithm 1 shows the

Table 2
Sample from the dataset.

Word	Normalized Word	Sense	Sense Translation	Sentence	Sentence Translation
اب (>b)	اب	والد الشخص.	Father	كشفت دراسة جديدة أن عناق الأب مهم جدا.	A new study has revealed that father's hug is very important.
اب (>b)	اب	الشهر الثامن من السنة	August	حسب الحسابات الفلكية، فسيحل شهر رمضان يوم الخميس 10 آب/ أغسطس الجاري.	A new study has revealed that father's hug is significant.
اب (>b)	اب	لقب كنيسي لرجل الدين المسيحي.	Church Father	تقلد الأب مانويل وظائف ومسؤوليات دينية وسياسية عديدة.	A new study has revealed that father's hug is significant.

Table 3
Comparison between the collected dataset and the dataset published by El-Razzaz et al. (2021).

Word	Sense	Sense Translation	Sample from proposed Dataset	Sample from El-Razzaz et al. (2021) Dataset
ألم (>alam)	وجع شديد.	Ache	تعد مسلكيات الألم مثل البارا سيتامول ومضادات الالتهاب غير السيتيروييدية ... ومعرفة الآثار الجانبية التي قد تسببها.	، - ألم وآلام.
دهن (dahon)	دهن الجدار ونحوه: طلاؤه.	Painting	يتم دهن الجدار إما بفرشاة الدهان أو الاستعانة بمكينة رش لتغطية أفضل.	:- مصدر دهن.

algorithm used for data collection.

Algorithm 1 Data collection and annotation approach

Input: URL to scrape, *u*
Set of polysemous words, *p*
Output: Set of sentences, *s*
import the scraping library
for each polysemous word, p in *p* **do**
use the scraper to download content from website u
extract text that contains the polysemous word p from the content
add extracted sentences with corresponding polysemous word p to s
end for
check s manually
assign correct senses to s based on the dictionary
check for missing data
perform a web search to ascertain appropriate sentences for missing senses
use GPT3.5 to fill in missing data
check GPT3.5 generated data manually

4.2. Comparison with other datasets

The dataset set out in this paper has been systematically collected to demonstrate high completeness and systematicity. This intended approach makes it distinct from other current datasets with more polysemous words but lacks systematicity and completeness in their samples. The reason behind the following approach provided in section 4.1 is as follows:

- The completeness of sentence examples per sense maintains contextual coherence and offers a more representative depiction of natural language usage. Incomplete sentences, often present in other datasets, may introduce ambiguity and noise into the dataset.
- The consistency in sample size for every sense of the target words is pivotal for meaningful and unbiased evaluation of WSD models. Datasets with varying example counts for different senses can lead to biased sense decisions.

The approach followed for data collection shows its comprehensiveness despite consisting of 100 polysemous words. This comprehensiveness has made it a valuable resource for WSD tasks. Distinct from other Arabic WSD datasets, it comprises complete sentences as examples for each sense. Moreover, the data is sourced from various domains to build diverse data and reduce bias when training WSD models. Comparing the dataset presented in this paper with the publicly available dataset published by El-Razzaz et al. (2021) will show how valuable the collected data is for Arabic WSD research. The comparison is made with El-Razzaz et al. (2021) since it is the only data available online. El-Razzaz *et al.* data is large yet not comprehensive. A WSD system trained on this data will not learn different word senses efficiently. It includes incomplete sentences as examples for some senses. Besides, it is built to have a single correct example corresponding to a sense. Therefore, the WSD system will not learn different word senses effectively, and bias will be introduced. Table 3 compares the proposed data with El-Razzaz *et al.* data, showing the comprehensiveness of the proposed dataset. As shown in the table, some examples in the ElRazzaz *et al.* dataset include two words that do not compose a complete sentence and are not considered beneficial to extract contextual information for a given sense.

4.3. Statistical analyses

Different statistical analyses have been carried out to improve the dataset's clarity and readability. These include examining the distribution of senses per word, assessing the richness and diversity of the vocabulary in the dataset, measuring the proportion of content words that possess meanings, evaluating the distribution of POS, analyzing the word categories, and studying the POS associated with different senses.

4.3.1. Distribution of senses per word

The resulting dataset comprehensively encompasses multiple senses associated with each polysemous word, resulting in 3670 instances. The data collection process provides a distinctive and representative set of sentences that exhibit the various meanings of words in diverse contextual settings. Table 4 summarizes the distribution of polysemous words, their associated senses, and the total examples provided per word. This table also offers a comparative analysis between the dataset introduced in this paper and the dataset proposed by El-Razzaz et al. (2021), which is the only dataset available online.

The ratio of total examples per word to the number of senses per word is used to measure the discrepancy of El-Razzaz et al. (2021) dataset. This ratio reveals that the proposed dataset provides a more diverse and contextually rich collection of examples, with each sense being exemplified by ten sentences. Conversely, in the dataset created by El-Razzaz et al. (2021), each sense is represented by a singular sentence example, which is insufficient for the model to understand context and meaning effectively. Furthermore, as elucidated in Table 3, these sentences within the El-Razzaz et al. (2021) dataset are occasionally fragmentary and incomplete. This shows that the dataset lacks comprehensiveness.

4.3.2. Corpus statistics

The resulting dataset is a collection from multiple resources to get more diverse and extensive data. Table 5 summarizes the corpus statistics considering the origin where samples are collected, the number of tokens, unique tokens, and context sense pairs. It shows the vocabulary diversity and richness exhibited by the text from each source for handling polysemy. The 'Number of Tokens' in each source represents the volume of textual data, with larger values for sources like "Other sources," "Aljazeera," and "Arabia Weather." The 'Number of Unique Tokens' denotes vocabulary richness, with sources like "Aljazeera Documentary," "Argaam," "CNN Arabia," and "Wikipedia" showcasing the highest proportion of unique tokens. The reliability of sources, such as "Wikipedia" and "Aljazeera," and the specificity of content, as in "Altibbi" and "Arabia Weather," are crucial considerations. They offer well-structured, high-quality content containing general and domain-specific polysemous words that could be beneficial for generalizing WSD tasks related to those domains.

Table 6 presents a comparative analysis of the total number of tokens within the dataset introduced in this paper and the El-Razzaz et al. (2021) dataset. While the El-Razzaz et al. (2021) dataset is notably more extensive in data size, containing much more polysemous words, it falls significantly short regarding the total and unique tokens it encompasses. The total number of tokens and the count of unique tokens in the proposed dataset are approximately double that found in the El-Razzaz et al.

Table 4
Word senses distribution per target polysemous word.

Number of Senses Per Word	Proposed Dataset			El-Razzaz et al. (2021) Dataset		
	Total Examples Per Word	**Total Word Count Per Sense**	**Ratio of total examples per sense**	**Total Examples Per Word**	**Total Word Count Per Sense**	**Ratio of total examples per sense**
3	30	57	10	3	1110	1
4	40	30	10	4	512	1
5	50	6	10	5	256	1
6	60	4	10	6	132	1
7	70	2	10	7	67	1
8	80	1	10	8	52	1

Table 5
Word senses distribution per target polysemous word.

Source	Number of Tokens	Number of Unique Tokens	Number of Context sense pairs
Argaam	719	494	101
Aljazeera Documentary	840	590	57
CNN Arabia	906	619	102
BBC Arabia	1764	1056	193
Shifaa	3243	1744	340
Wikipedia	4789	2769	580
Arabic Post	6528	3726	736
Altibbi	7263	3485	739
Arabia Weather	13,131	6198	1251
Aljazeera	40,884	16,277	4375
Other sources	50,254	20,491	6008

Table 6
Comparison between total and unique tokens within the proposed and El-Razzaz et al. (2021) dataset.

Dataset	Total Number of Tokens	Total Number of Unique Tokens
Proposed Dataset	130,321	57,449
El-Razzaz et al. (2021) Dataset	69,134	25,640

(2021) dataset. In WSD, the data quality, especially regarding token variety and diversity, is crucial. The proposed dataset takes the lead in this context, offering a richer vocabulary and a more comprehensive understanding of word meanings. For machine learning models that rely on these datasets, diverse and extensive vocabulary availability is paramount, ensuring that they can effectively grasp the nuances of language and semantics.

4.3.3. Vocabulary density

The Vocabulary Density (VD) metric is used to analyze and understand the WSD data. This metric measures the richness and diversity of the vocabulary within a dataset. It can provide information about the dataset's completeness and level of ambiguity. Two variables are required for calculating VD: the number of unique words and the overall number of words. The general equation of VD is presented in (1):

$$VD = \frac{uw}{tw} \tag{1}$$

where uw is the number of unique words, and tw is the total number of words.

The generated data is composed of a hundred words, each with multiple senses. First, the VD is calculated per sense, then the average value per all senses is evaluated to compute the VD value per word. The per sense calculations are found in (2):

$$VD_s = \frac{uw_s}{tw_s} \tag{2}$$

where uw_s is the total number of unique found in the total sentence examples in a given sense s and tw_s is the total number of words in all sentences in a sense s. The average VD per all senses is displayed in (3).

$$VD_{pw} = \frac{\sum_{i=1}^{n} VD_{s_i}}{|S(w_i)|} \tag{3}$$

where VD_{s_i} is the VD value for word s in a sense i and $|S(w_i)|$ is the total number of senses of the word i.

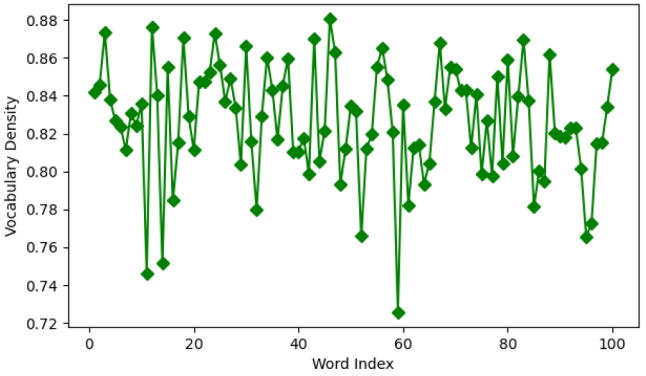

Fig. 1. Vocabulary density for the collected dataset.

Fig. 1 shows the VD value per polysemous word. It ranges between 0.72 and 0.88, meaning there are many unique words due to the relatively high vocabulary density. This VD range indicates linguistic diversity in this data, suggesting moderate ambiguity. Word disambiguation may require careful analyses and consideration to differentiate between distinct word senses accurately since this category has many different unique words or senses.

Fig. 2 displays the VD of the El-Razzaz et al. (2021) dataset. In this context, VD values vary between 0.5 and 1. A VD of 0.5 indicates that half of the words are unique for a given set of words, and the other half are repetitions of words already present. A VD of 1 suggests that all the words in each context are unique. However, some context samples in this dataset consist of only two words. Even if all the words are unique, the context might not be linguistically diverse or meaningful. Therefore, a VD of 1 does not necessarily reflect significant linguistic diversity in such situations.

4.3.4. Lexical density

The Lexical Density (LD) measurement was carried out to study the richness and clarity of the dataset. This metric allows understanding the distribution of content words (i.e., nouns, verbs, adjectives, and adverbs) in different senses of polysemantic words. LD provides insight into how distinct and informative examples are for individual senses. The computation is done by measuring the number of content words concerning the total number of words within the text of a single polysemous word. It will provide valuable information about the nature of the WSD dataset and its possible impact on disambiguating words. The LD calculation is performed based on a formula presented in (4), with the first step consisting in measuring LD per sense:

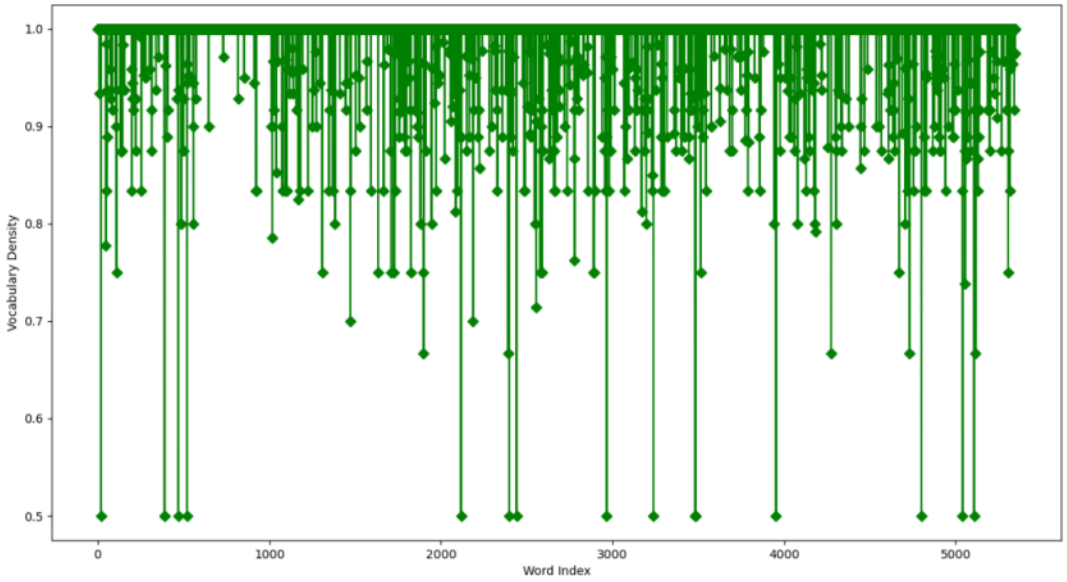

Fig. 2. Vocabulary density for El-Razzaz et al. (2021) dataset.

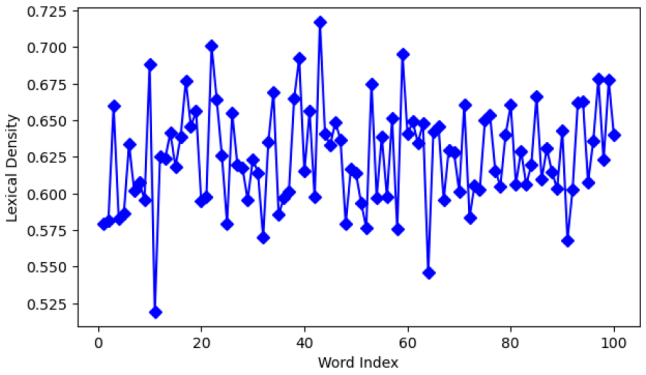

Fig. 3. Lexical density for the collected dataset.

$$LD_s = \frac{cw_s}{tw_s} \tag{4}$$

where cw_s is the total number of content words found in the total sentence examples in a given sense s and tw_s is the total number of words in all sentences on a sense s. Then, LD per polysemous word is measured using (5).

$$LD_{pw} = \frac{\sum_{i=1}^{n} LD_{s_i}}{|S(w_i)|} \tag{5}$$

where LD_{s_i} is the LD value for word s in a sense i and $|S(w_i)|$ is the total number of senses of the word i.

Fig. 3 depicts the LD for each polysemous word in the dataset. It shows that the LD value ranges between 0.519 and 0.717. It indicates that the sense being considered has a moderate to high amount of information-rich content words. The moderate LD value could indicate that the sense is relatively informative or contains substantial, meaningful content for disambiguation. A higher lexical density implies that the examples contain more relevant content words, which can potentially aid in distinguishing between different senses of a word.

Fig. 4 presents the LD distribution of Arabic polysemous words

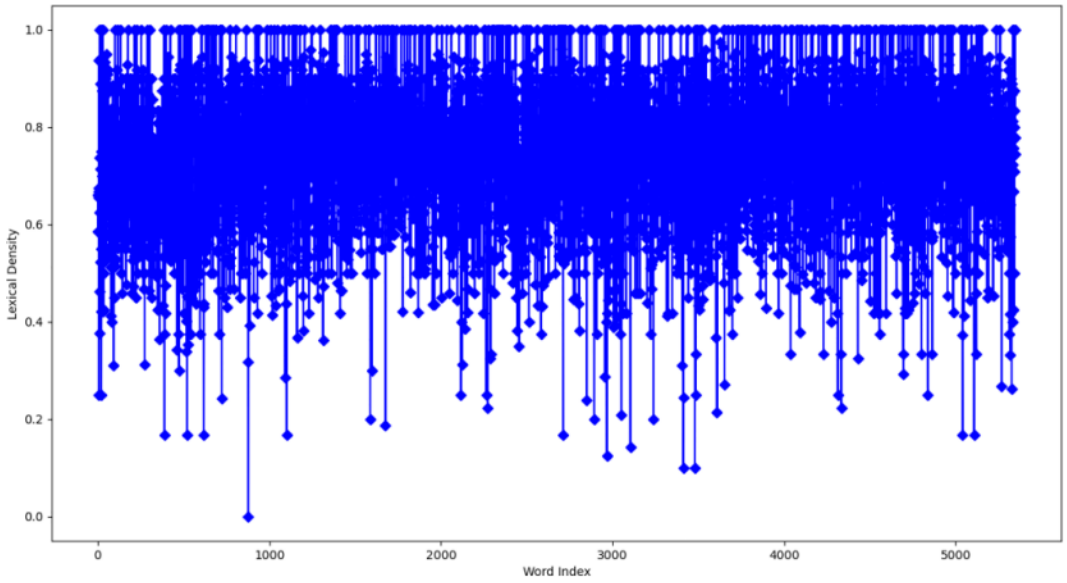

Fig. 4. Lexical density for El-Razzaz et al. (2021) dataset.

within the El-Razzaz et al. (2021) dataset. The Figure shows that the LD ranges from 0 to 1. This variation captures the variability in LD, signifying the proportion of content words to the total number of words across instances. Instances with lower LD values indicate a prevalence of function words, potentially lacking informative content crucial for effective WSD. Conversely, instances with higher LD values suggest a denser information load, posing challenges for the model in discerning subtle differences in word sense.

In contrast to the El-Razzaz et al. (2021) dataset, the proposed dataset in this paper demonstrates a more balanced dataset, making it more suitable for the WSD task. Consistently maintaining a moderate to high lexical density, the proposed dataset is well-suited for training models that aim to capture both the complexities of language and the distinctions in word senses. These characteristics make the proposed dataset more optimal for achieving effective WSD outcomes.

An extensive dataset with a hundred polysemantic words is crucial for implementing WSD systems. Even though other datasets are larger, they often lack comprehensiveness. These resources may exhibit fewer instances per sense and an uneven distribution across different meanings, potentially leading to biases in the model's learning process. The manually curated dataset of one hundred polysemantic words addresses these concerns by prioritizing depth and balance. It is collected to ensure an even distribution of instances among different senses and prevents the model from favoring the majority of samples. Therefore, the emphasis should be on the dataset's quality, diversity, and balance rather than solely on the quantity of words.

4.3.5. Part of speech

The overall distribution of the POS for each sense in the dataset is presented in Fig. 5. The dataset predominantly comprises noun types, with verbs as the second most frequent POS. Fig. 5 shows that approximately 60 % of the senses correspond to nouns, while 31 % are verbs. The remaining 9 % of senses are distributed among proper nouns, adjectives, adverbs, and negative particles. Following the dataset's observed importance of noun senses, Arabic is characterized by a significant degree of dominance in sense variations due to changes in POS. Most of the polysemous words appear to be nouns in this dataset. This variation does not affect the performance of the model as the sense variation appears in both nouns and verbs at most.

Fig. 6 presents the distribution of POS tags in the El-Razzaz et al. (2021) dataset. In this dataset, nouns are the most prevalent, followed by verbs as the second most frequent POS tag. Approximately 52 % of the dataset's senses are nouns, while 35 % are verbs. This distribution closely aligns with the POS tag distribution in the collected dataset, as shown in Fig. 5. The observed pattern suggests that polysemy in Arabic words is primarily found in nouns and verbs.

Fig. 6 introduces additional POS tag categories compared to Fig. 5, encompassing vocative particles, future particles, interrogative particles, focus particles, particles, prepositions, and subordinating conjunctions. These POS types were excluded from the collected dataset, where the focus was on comprehensive data for the most common and widely used MSA words. The dataset presented in this paper emphasized identifying polysemy in content words, as they significantly impact overall ambiguity by conveying the primary meaning of a sentence. Certain function words were intentionally excluded. These words include future particles ("ألا" - that; indeed; oh; truly; lest; verily), emphatic particles ("إن" – if, indeed, when), vocative particles ("يا" - hey), interrogative particles ("ما" - how much; which), focus particles ("أما" - but, however, concerning, indeed), subordinating conjunctions ("بينما" - while, during, although), and prepositions ("إلى" - to, for). These function words primarily contribute to grammatical structure and relationships between terms. While polysemy in function words may contribute to ambiguity, its impact is generally more limited than in content words. The data presented in this paper specifically focused on instances of polysemy in function words when the word is most common and exhibits an ambiguity involving both content and function words.

4.3.6. Word category

When dealing with MSA, it should be noted that each word sense is associated with a particular category, like sociology, physiology, anatomy, and others. Consequently, manual labeling of the word categories was carried out. The distribution of each category in the dataset has been presented in the bar plot depicted in Fig. 6, arranged in a descending order. It is observed that the category of "act" emerges as the most prevalent category among the senses. This domination is also illustrated by Fig. 3, as the noun and the verb are the most prevalent POS tags in the dataset. For example, the word "ترك" exemplifies this phenomenon, as it can be both a noun meaning to leave pronounced as (tarok) or a verb meaning left and pronounced as (taraka) while both are conveying the notion of action.

Fig. 7 demonstrates that the second contributing category is general. In the dataset, the general category is added when a single sense can be used in multiple contexts, and each refers to a specific category. For example, the word "جوهر" (jawohar) is an adjective that is used to

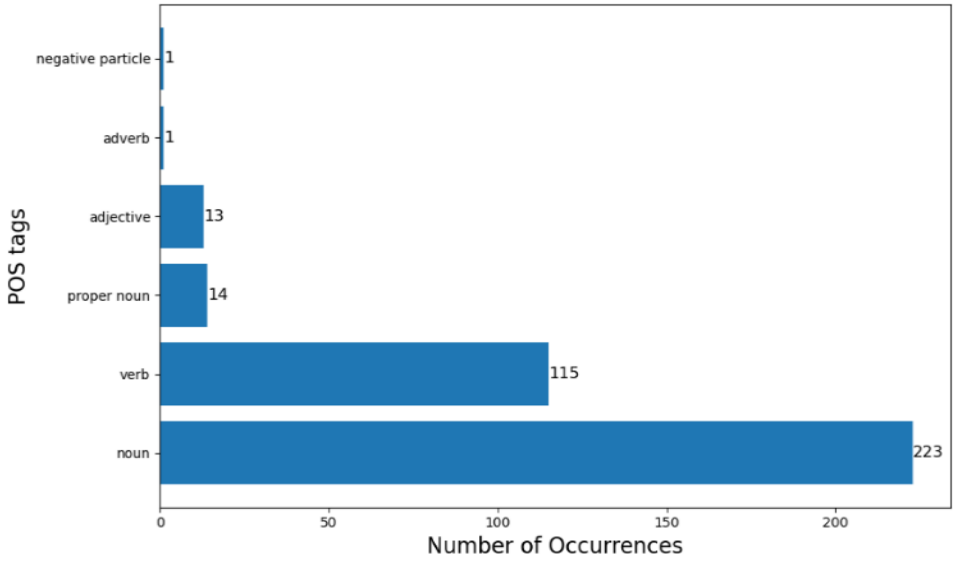

Fig. 5. POS tag distribution for all senses presented in the dataset.

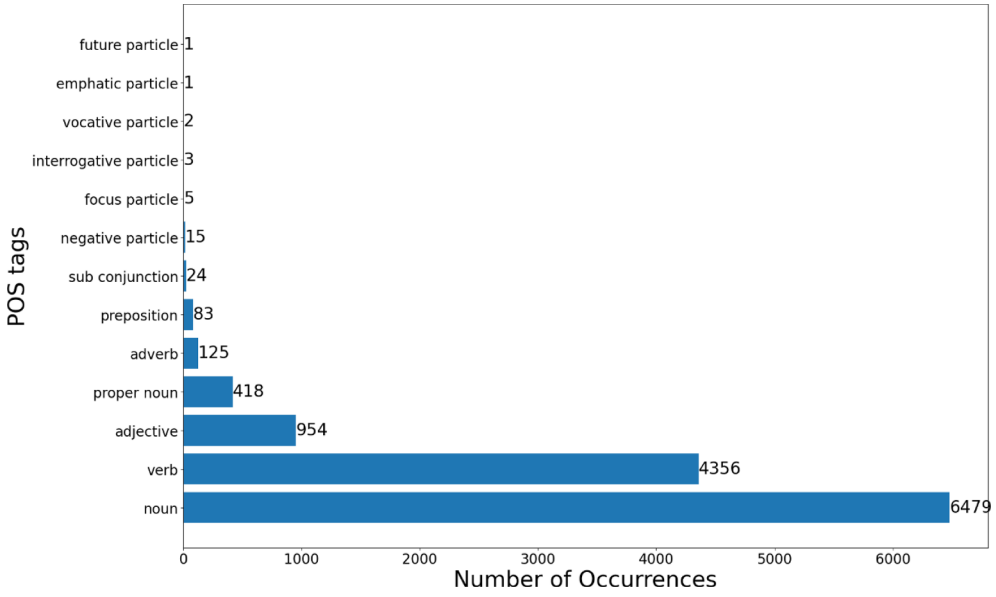

Fig. 6. POS tag distribution for all senses presented in El-Razzaz et al. (2021) dataset.

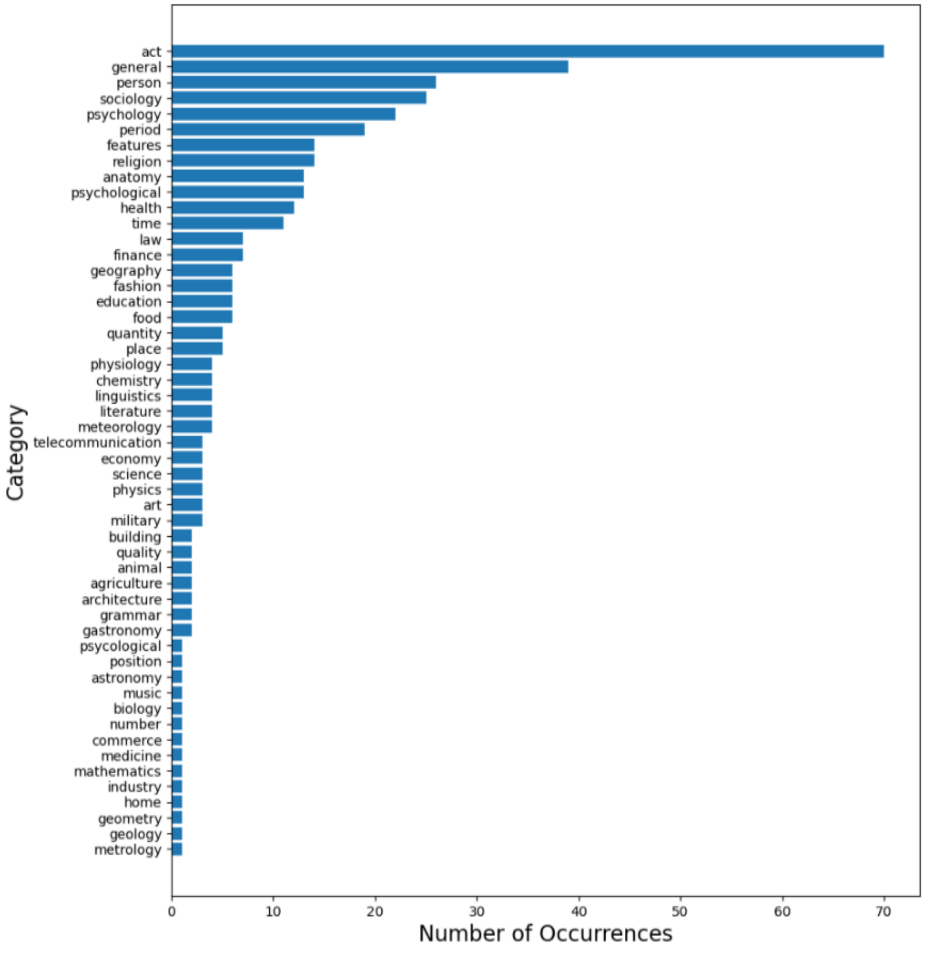

Fig. 7. Category distribution for all senses presented in the dataset.

describe something unique or beautiful. However, it might be describing a place (geography category), jewelry (fashion category), or building (architecture category). Thus, it belongs to the general category because it might be linked to multiple categories.

5. Classification approach for word sense disambiguation

The classification task for WSD is considered convenient when labeled data is available for training. Classification of word senses will allow the model to learn distinctions and contextual cues among various

word senses. The preference for a classification model over clustering is particularly evident when certain senses exhibit shared contexts. Consider the Arabic word "فصل," (fSl), which can refer to a semester, a classroom, or a book chapter. These three senses are closely related, potentially sharing similar contextual words. Opting for clustering in such cases may inadvertently introduce false positives, as clustering tends to group instances based on similarity, which might not align with the specific distinctions required in WSD. Therefore, a classification approach is more suitable for capturing distinct senses, ensuring a more accurate and context-aware disambiguation process.

The disambiguation of words encompasses several consecutive steps. It begins with preprocessing Kaddoura et al. (2021), data preparation, feature extraction, and sense recognition. Diverse embedding techniques exhibit tailored efficacy in addressing specific tasks or domains. Word2Vec and FastText embeddings prove exceptionally adaptable when dealing with limited training data, showcasing their ability to extract meaningful representations from sparse information. In contrast, BERT can capture intricate contextual nuances, demonstrating remarkable resilience by sustaining high performance even with smaller training datasets. Hence, BERT possesses a distinctive quality that distinguishes it from methodologies reliant on extensive data for optimal results. The significance of this distinction becomes evident when considering related studies. Alkhatlan, Kalita, and Alhaddad (2018) achieved noteworthy results using skip-gram and GloVe, with a maximum accuracy of 82.17 % for SkipGram and 71.73 % for GloVe. Simultaneously, in the investigation by Alian and Awajan (2020), FastText displayed its effectiveness with a maximum accuracy of 46.6 %. These outcomes underscore the nuanced effectiveness of various embedding techniques. However, they also signal an opportunity for improvement, prompting the selection of BERT for the present research.

Another advantage of utilizing BERT for Arabic WSD is its ability to comprehend the Arabic language's complex contextual and bidirectional aspects. BERT's proficiency in understanding the context in which words operate extends to multiple languages, making it a versatile choice for Arabic WSD. BERT offers the flexibility of fine-tuning Arabic-specific WSD datasets, thereby enabling the adaptation to the nuances of the Arabic language. The transfer learning approach circumvents the necessity for extensively annotated corpora to enhance the development of Arabic WSD models. Given the scarcity of comprehensive WSD corpora for Arabic, this approach is beneficial. Previous approaches, including those presented by Elmougy et al. (2008), such as Naive Bayes (NB), did not achieve a high accuracy level of disambiguation, 75.80 %. Pre-trained transformer models consistently outperform other classical algorithms in various NLP applications.

Thus, in this paper, the BERT model is applied to context sense pairs to classify the proper sense for ambiguous words. The pre-trained transformer model, BERT, is trained on a WSD-specific dataset containing context, word senses, and meanings. BERT parameters are fine-tuned to align with language complexities, enhancing performance and accuracy in disambiguating word senses within context.

Several data preparation steps are proposed to enhance the performance of the suggested approach for disambiguation of the Arabic language.

5.1. Preprocessing

Most individuals do not use diacritics in their text. Although diacritics improve readability, they are not essential as human readers can easily comprehend the text. The absence of diacritics in most written or published Arabic articles is a significant reason for increasing linguistic ambiguity Kaddoura et al. (2022). Moreover, ambiguation increases with the normalization of most published texts and the agglutinative nature of Arabic. Data preprocessing has been carried out to allow efficient learning of the senses. Firstly, diacritics were removed from the data since some text found on the web includes diacritics. This step aimed to ensure systematic data since most samples in the dataset do not include diacritics. Furthermore, normalization was carried out to enhance the learning process of the model, transforming "إأآا" into "ا" for consistency. Regarding the agglutinative nature, the AraBERT Antoun et al. (2020) tokenizer has been selected to tokenize the samples, transforming these words into multiple words. Fig. 8 illustrates the tokenization process. The agglutinative word is segmented into multiple tokens through the separations of pronouns like "وهم" and conjunctions like "و" from the target word. Algorithm 2 illustrates the implementation process for the preprocessing of Arabic words.

Algorithm 2 Data preprocessing

Input: set of sentences containing target words, *s*
set of senses for target words, *g*
Output: cleaned sentences tokens, *cs*
cleaned senses tokens, *cg*

```
import the Arabic letters and text manipulation library
import bert_tokenizer
for each sentence, s in S do
    cleaned_sent ← Φ
    for each word, w in S do
        cleared_word ← transform "إأآا" into "ا"
        cleared_word ← remove diacritics
        append cleared_word to cleaned_sent
    end for
    tokens ← bert_tokenizer(cleanedsent)
    append tokens to cs
end for
for each sense, g in G do
    cleaned_sense ← Φ
    for each word, tw in G do
        cleared_tw ← transform "إأآا" into "ا"
        cleared_tw ← remove diacritics
        append cleared_tw to cleaned_sense
    end for
    tokens_g ← bert_tokenizer(cleaned_sense)
    append tokens to cg
end for
```

As for root extraction, due to an ongoing problem with Arabic language processing in which there is a lack of accurate Arabic stemmers/lemmatizers that could correctly extract word roots, it has not been possible to perform root extraction for all words. This limitation is essential, as incorrect stems can lead to different appearances of words, making it difficult for the model to learn correctly. Table 7 presents an example of stemming the words "يرمي" (yaromy) and "فأسقيناكموها" (fa > soqayonA-kumwhA) using different stemmers. As demonstrated in the table, altering the word resulted in inaccurate stemming results. This limitation may adversely affect models, mainly when the terms "رمى"

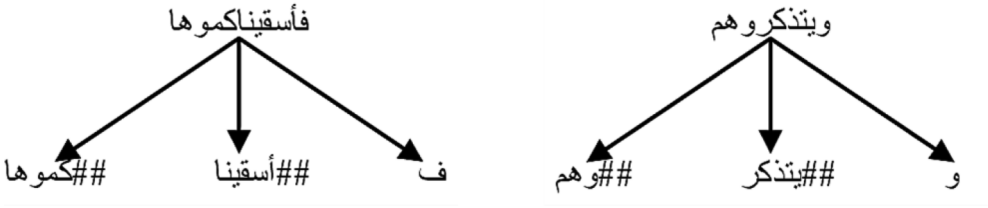

Fig. 8. Tokenization examples.

Table 7
Example of word stemming using state-of-the-art stemmers.

Word	ISRI stemmer	Snowball stemmer	Lancaster stemmer	Arabic LightStemmer	Farasa	CAMel	Correct Root
يرمي فأسقيناكموها	يرم فأسقيناكمو	يرم اسقين	يرمي فأسقيناكموها	رمى فأسقيناكموها	رمى فأسقيناكموها	رمى اسقيناكموها	رمى سقى

Table 8
Percentage of error per stemmer.

Stemmer	Percentage of error
ISRI stemmer	26.32
Snowball stemmer	28.01
Lancaster stemmer	50.35
Arabic LightStemmer	35.08
Farasa	10.48
CAMel	14.93

(ramY) and "سقى" (saqY) are used for disambiguation or analyses. Thus, the proposed model will not learn incorrect words by avoiding returning words to their root format. For this reason, stemming was not implemented in this study.

To demonstrate the impact of stemming on model performance and how it will influence contextual information due to variations in word forms, an evaluation of the stemmers presented in Table 8 is conducted using the dataset provided in this paper. The error percentages reveal that none of the stemmers effectively extracted the root forms of the polysemous words. The table shows that the Farasa stemmer is the most accurate, with the lowest error percentage of 10.48 %. However, although the percentage of error is considered low, wrong stem words will negatively affect the WSD model. To illustrate, an incorrect word stem will change the contextual embeddings of a given context, thus making it similar to a different word, increasing the disambiguation errors.

5.2. Feature extraction

Deep learning models can understand data only in numerical format; thus, numerical features must be extracted before training the AraBERT Antoun et al. (2020) model. The AraBERT model was selected over other pre-trained models because it consistently outperformed them. According to Al-Hajj and Jarrar (2022) and Saidi et al. (2023), AraBERT is considered the best-performing model for WSD. First, the data must be reshaped into sentences, sense pairs, and corresponding labels. All available senses are extracted for each polysemous word in the dataset, and combinations are made with each sentence and all senses. A zero label is assigned if the sense is incorrect, whereas one label is assigned for the correct sense. Two data shapes have been used. Table 9 illustrates a sample for a single sense for the polysemous word "اب" (>b) for the data. It represents the data for a simple sequence classification task using BERT. The data presented in Table 9 is called a "sentence-sense dataset". Considering the sentence and sense pairs, the dataset comprises 14,482 samples.

The polysemous target word is annotated to improve the model performance by enclosing it within double quotation marks. The annotation is achieved by identifying the index of the target word within the sentence data and placing double quotation marks before and after it. Moreover, to imbue the senses with weak supervision, the target word is added before its corresponding meaning in the glosses. Table 10 depicts samples from sentence-sense pairs with weak supervision data. The data presented in Table 10 is referred to as a "sentence-sense with weak supervision dataset.".

The Arabic WSD using BERT is a sequence classification task. Thus, each sample (sentence and sense/sense pair) in the dataset is combined using a '[SEP]' token before converting the sequence into features. In BERT, sentences are converted into tokens, and a '[CLS]' token is added at the beginning and a '[SEP]' token at the end. Fig. 9 illustrates a sample from sentence-sense after preprocessing and before conversion to features. It presents transforming sentences and senses into tokens and combining them into a single sequence.

Next, the tokens are converted into features. Four BERT features are extracted: (1) input_ids, where the tokens are converted into IDs, also known as contextual embeddings. (2) attention mask, which consists of all ones and has a size equal to the length of the BERT tokens. (3) token type IDs, where "1″ is added for indices where the gloss is present in the sequence. (4) the label. After the data is converted into features, the maximum sequence length is 256. Consequently, zero padding is added to all features except the label, either 0 or 1, to ensure that all input features have the same length; if the sequence length exceeds 256, it will be truncated. Fig. 10 presents an example of input features for the data sample presented in Fig. 9. In this sample, token_type_ids are set to "1″ at the index of the sense.

Additional features have been concatenated with the BERT model to ensure the robustness of the WSD model. These features include POS tags and frequency counts, which will serve as a base to measure the level of disambiguation features. As for POS tags, as shown in Fig. 5, the polysemous word senses are divided into six tags (noun, verb, proper noun, adjective, adverb, and negative particle). The POS tags are first converted to IDs to incorporate them as features with the BERT model because deep learning models can only understand numerical data. Then, a new feature named "target_mask" consisting of the POS tag ID is introduced. The target_mask consists of all zeros, and the POS tag id is placed at the index of the target word in the sequence. As for frequency count, it is concatenated with the sequence of sentences and senses for adding emphasis. So, this frequency count feature is presented in the input_ids, and for word annotation, it is also added in the target mask. Fig. 11 illustrates the target mask for POS and frequency count.

5.3. Sense recognition

The AraBERT model Antoun et al. (2020) for sequence classification is used for sense recognition. The AraBERT model by Antoun et al. (2020) is fine-tuned by setting the maximum length to 256. The BERT model decides whether the sense corresponds to the sentence. The dataset is split into training, validation, and testing by a proportion of 80:10:10. To ensure that all senses for each polysemous word exist in all trains, valid and test data are split systematically. The first 80 % per each sense is taken for training, the following 10 % for validation, and the last 10 % for testing. To address WSD, BERT is used as a base model and trained multiple models, each with its own set of distinct and additional features. Training multiple models will help know the linguistic features or context contributing to sense disambiguation. Each model is evaluated on the test data samples, and then statistics are performed to effectively assess the model efficacy for disambiguating words. The proposed experiments are as follows: a baseline model for sequence classification using BERT, an algorithm presented in Algorithm 3, that incorporates (1) the POS tags, (2) word frequency count, (3) weak supervision, (4) weak supervision with the POS tag, (5) weak supervision with word frequency count, and (6) ensemble BERT.

5.3.1. Experiment 1 POS-BERT

A POS-BERT model enhances the baseline sequence classification BERT model when integrated with POS as an additional feature. In Arabic, POS is essential for understanding words with multiple meanings. Including a POS tag as part of BERT would be beneficial in

Table 9
Sample from the sentence-sense dataset.

Word	Example Sentence	Sentence Translation	Sense	Sense Translation	Label
اب(>b)	اكتشفت دراسة جديدة أن عناق الأب مهم جدا.	A new study has revealed that father's hug is very important.	والد الشخص.	Father	1
اب(>b)	اكتشفت دراسة جديدة أن عناق الأب مهم جدا.	A new study has revealed that father's hug is significant.	الشهر الثامن من السنة	August	0
اب(>b)	اكتشفت دراسة جديدة أن عناق الأب مهم جدا.	A new study has revealed that father's hug is significant.	يحيي المسيحيون الذين لجأوا لكنيسة بقبلة.	Church Father	1

Table 10
Sample from sentence-sense pairs with weak supervision data.

Word	Example Sentence	Sentence Translation	Sense	Sense Translation	Label
اب(>b)	اكتشفت دراسة جديدة أن عناق "الأب" مهم جدا.	A new study has revealed that father's hug is very important.	والد الشخص.	Father	1
اب(>b)	اكتشفت دراسة جديدة أن عناق "الأب" مهم جدا.	A new study has revealed that father's hug is significant.	الشهر الثامن من السنة.	August	0
اب(>b)	اكتشفت دراسة جديدة أن عناق "الأب" مهم جدا.	A new study has revealed that father's hug is significant.	يحيي المسيحيون الذين لجأوا لكنيسة بقبلة.	Church Father	1

Sentence: كشفت دراسة جديدة أن عناق الأب مهم جدا
Gloss: والد الشخص
Input Sequence: '[CLS]', 'كشفت', 'دراسة', 'جديدة', 'أن', 'عناق', 'الأب', 'مهم', 'جدا', '[SEP]', 'والد', 'الشخص',

Fig. 9. Preprocessing and sequence formation.

Input_ids: [2, 5278, 2856, 1252, 331, 49327, 2614, 3581, 3, 1913, 53268, 3, 0, 0, ….., 0, 0]
Attention_mask: [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, ….., 0, 0]
Token_type_ids: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, ….., 0]
Label: 1

Fig. 10. Features extracted.

POS: Noun
POS_id: 2
Target_mask: [0, ,0, 0, 0, 0, 2, 0, ….., 0, 0]
Freq = 117
Input_ids: [2, 5278, 2856, 1252, 331, 49327, 2614, 3581, 3, 1913, 53268, 3, 117, 3, 0, ….., 0, 0]
Target_mask: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 117, 3, 0 ….., 0]

Fig. 11. POS and frequency feature extraction.

disambiguating polysemous words since it directly corresponds with the sense of the word. Integrating POS tags into BERT will allow the model to understand word senses better, leveraging a direct correlation between POS information and word meaning. POS would be beneficial if the polysemous word sense is based on its grammatical context. A target mask for a POS tag feature, combined with BERT model features to display an annotation for the target word's presence in context and its POS, can be seen in Fig. 11. This word representation will enable the WSD model to effectively disambiguate word senses by leveraging semantic and syntactic cues. The implementation of POS-BERT is presented from lines 11 to 17 in Algorithm 4.

Algorithm 3 Baseline BERT model for WSD sequence classification

Input:	cleaned sentences tokens, *cs*
	cleaned senses tokens, *cg*
Output:	label, *l* {0,1}
	set of percentage of error per word, *pe*
	average percentage of error, *ape*
	number of errors per word, *ne*
1:	import the required modules
2:	*bert_tokens* ← concatenate *cs* and *cg* using the 'SEP' token
3:	insert 'CLS' token at the beginning of *bert_tokens*
4:	insert 'SEP' token at the end of *bert_tokens*
5:	*input_id* ← obtain contextual embeddings for *bert_tokens*
6:	*attention_mask* ← create a list of ones with the length of *bert_tokens*
7:	*token_type_id* ← create a list of zeros with the length of *cs* + 2
8:	*token_type_id* += create a list of ones with the length of *cg* + 1
9:	pad *input_id*, *attention_mask*, *token_type_id* with zeros to a length of 256
10:	run the BERT model on the padded inputs
11:	predict word senses for the test data
12:	*pe* ← calculate the percentage of error per target word in the test data
13:	*ape* ← calculate the average percentage of error for the entire test data
14:	*ne* ← count the number of errors per target word in the test data

5.3.2. Experiment 2 Frequency-BERT

Including word frequency count in the baseline BERT model will help improve the predictions for WSD. The integration of frequency feature is chosen due to Zipf's principle, which significances the relation between word frequency and the number of meanings Zipf (1945). The Frequency-BERT introduces a quantitative measure for disambiguation, where it balances contextual information and the frequency of rare words. By utilizing this feature, the frequency of polysemous words appearing in Arabic may be deduced, and the relevant impact on their inherent misunderstandings can be observed. Algorithm 4 depicts how the frequency count feature is integrated with BERT features. Its implementation is presented in lines 18 to 27 in Algorithm 4.

5.3.3. Experiment 3 weak supervision

In the literature Huang et al. (2019), weak supervision is introduced to improve the performance of the baseline BERT model for sequence classification tasks. It facilitates the annotation of the target word and its associated sense. By incorporating weak supervision, the model's ability to correctly identify and classify the target polysemous word in context is enhanced, ultimately leading to improved overall performance. Using weak supervision complements the existing BERT model's capabilities. It allows the use of additional information and knowledge sources to analyze target words and their corresponding meaning better, which results in more precise and effective sequence classification. Algorithm 4 presents the BERT model with weak supervision in lines 28 to 31. It is very similar to the baseline BERT model; however, the difference is in the dataset shape.

5.3.4. Experiment 4 weak supervision POS-BERT

A combination of POS-BERT with weak supervision is essential in improving the accuracy and effectiveness of sequence classification tasks. POS-BERT influences the power of the baseline BERT model and the inclusion of POS tags, which serve as valuable indicators for disambiguating polysemous words. This integration enables a more morphological understanding of word senses and improves the model's ability to annotate the target word and its corresponding sense accurately. This combination combines the strengths of POS-BERT and weak supervision, resulting in a more robust and efficient WSD system for classifying sequences to demonstrate improved accuracy in categorizing target words according to their different contexts. POS-BERT with weak supervision implementation is depicted in Algorithm 4 from line 32 to 35.

5.3.5. Experiment 5 weak supervision Frequency-BERT

The model better understands the relationships between word senses and their corresponding frequencies by incorporating word frequency count with the weak supervision framework. This combination allows

the model to capture better the morphological distinctions and disambiguation cues present in the data, thereby enhancing the robustness of the BERT sequence classification task. By leveraging both word frequency count and weak supervision, the model can make more informed predictions, leveraging the statistical regularities of word usage in context to improve the accuracy and precision of sequence classification. Ultimately, this integration empowers the model to achieve higher performance in disambiguating target words and assigning them to their appropriate senses. Algorithm 4 illustrates the implementation of Frequency-BERT with weak supervision in lines 36 to 39.

5.3.6. Experiment 6 ensemble BERT

Ensembling the previously mentioned models, namely POS-BERT, BERT with frequency count, weak supervision, POS-BERT with weak supervision, and frequency count BERT with weak supervision, will allow for improving the performance of the WSD system. Each model contributes unique features and techniques that address different aspects of the WSD problem. The strengths of individual models are merged by combining them through an ensemble. This ensemble approach works through weighted soft voting for the predictions of previously mentioned BERT-based approaches. It assigns higher weights for the more accurate models and thus exploits the shared knowledge and capabilities of these models. Algorithm 4 presents the ensemble BERT WSD algorithm.

6. Experimental results and analyses

6.1. Analysis of proposed models

The WSD BERT proposed models are included for comparison. Table 11 presents the F1 scores, the minimum and maximum number of errors per word, and the average error percentage for the implemented WSD models. The baseline BERT model with a frequency count reported the lowest F1- score, about 0.91. The lower score is because WSD is treated as a BERT sequence classification with no specific annotation for the target polysemous word. However, the F1-score remained high, indicating the ability to predict whether a sentence with a target word matches the suggested definition based on context.

Incorporating the POS tag with the baseline model resulted in an F1-score of 0.9167. POS tag provides information for different senses of a word and helps with disambiguation. Utilizing it as a feature will increase the F1-score above that of the model that incorporates frequency count as a feature. Further enhancement was achieved when annotating the target polysemous word using weak supervision. The model trained on sentence sense with weak supervision data and word frequency count reported the highest score, 0.95853. This is a considerable improvement in model performance, particularly for word disambiguation, because the weak supervision provides an annotation for the target word, and the frequency is an added feature that provides additional information to the model. The F1-score is further enhanced in ensemble BERT to reach 0.958687. Ensemble BERT is better at disambiguating words than any single BERT model because it has higher predictive accuracy based on a weighted majority vote.

Table 11
Results of the proposed BERT models.

Proposed Models	F1-score	Average percentage of error (%)	Min error value	Max error value	Number of words with zero error
Frequency-BERT	0.91045	7.09	0	6	45
POS-BERT	0.91674	6.42	0	5	54
Weak Supervision POS-BERT	0.94347	4.74	0	5	62
Weak Supervision	0.95653	3.81	0	3	67
Weak Supervision Frequency-BERT	0.95853	3.49	0	4	70
Ensemble BERT	0.95868	3.46	0	3	69

Algorithm 4 Weighted Voting Ensemble BERT for WSD Sequence Classification

Input:	cleaned sentences tokens, *cs*
	cleaned senses tokens, *cg*
Output:	Label, *l* {0,1}
	set of percentage of error per word, *pe*
	average percentage of error, *ape*
	number of error per word, *ne*
1:	import the required modules
2:	initialize weights for each BERT model: *pos_weight*, *freq_weight*, *ws_weight*, *ws_posweight*, *ws_freq_weight*
3:	*bert_tokens* ← concatenate *cs* and *cg* using the 'SEP' token
4:	insert 'CLS' token at the beginning of *bert_tokens*
5:	insert 'SEP' token at the end of *bert_tokens*
6:	*input_id* ← obtain contextual embeddings for *bert_tokens*
7:	*attention_mask* ← create a list of ones with the length of *bert_tokens*
8:	*token_type_id* ← create a list of zeros with the length of *cs* + 2
9:	*token_type_id* += create a list of ones with the length of *cg* + 1
10:	pad *input_id*, *attention_mask*, *token_type_id* with zeros to a length of 256
11:	**POS BERT Model**
12:	*target_mask_pos* ← create a list of zeros with a length of 256
13:	Find *target_position*
14:	Add POS tag embed in *target_mask_pos* at *target_position*
15:	*pos_logits* ← Run POS-BERT model on the padded inputs and *target_mask_pos*
16:	*pos_probabilities* ← *Softmax(pos_logits)*
17:	*pos_prediction* ← *argmax(pos_probabilities)*
18:	**Frequency BERT Model**
19:	*bert_tokens_f* ← concatenate *freq_count* to *bert_tokens* in line 5
20:	insert 'SEP' token at the end of *bert_tokens*
21:	Find *freq_count_position* in *bert_tokens_f*
22:	Add *freq_count* in padded *input_id* at *freq_count_position*
23:	*target_mask_f* ← create a list of zeros with a length of 256
24:	Add *freq_count* in *target_mask* at *freq_count_position*
25:	*freq_logits* ← Run Freq-BERT model on the padded inputs and *target_mask_f*
26:	*freq_probabilities* ← *Sof tmax(freq_logits)*
27:	*freq_prediction* ← *argmax(freq_probabilities)*
28:	**Weak Supervision BERT Model**
29:	*ws_logits* ← Run Weak Supervision BERT model on the padded inputs
30:	*ws_probabilities* ← *Sof tmax(ws_logits)*
31:	*ws_prediction* ← *argmax(ws_probabilities)*
32:	**Weak Supervision POS BERT Model**
33:	*ws_pos_logits* ← Run Weak Supervision POS-BERT model on the padded inputs and *target_mask_pos*
34:	*ws_pos_probabilities* ← *Sof tmax(ws_pos_logits)*
35:	*ws_pos_prediction* ← *argmax(ws_pos_probabilities)*
36:	**Weak Supervision Frequency BERT Model**
37:	*ws_freq_logits* ← Run Weak Supervision Frequency BERT model on the padded *inputs_target_mask_f*
38:	*w_freq_probabilities* ← *Sof tmax(ws_freq_logits)*
39:	*ws_freq_prediction* ← *argmax(ws_freq_probabilities)*
40:	**Weighted Voting**
41:	*pos_vote* ← *pos_weight* * *pos_probabilities[pos_prediction]*
42:	*freq_vote* ← *freq_weight* * *freq_probabilities[freq_prediction]*
43:	*ws_vote* ← *ws_weight* * *ws_probabilities[ws_prediction]*
44:	*ws_pos_vote* ← *ws_pos_weight* * *ws_pos_probabilities[ws_pos_prediction]*
45:	*ws_freq_vote* ← *ws_freq_weight* * *ws_freq_probabilities[ws_freq_prediction]*
46:	**Calculate the final label**
47:	*total_votes* ← *pos_vote* + *freq_vote* + *ws_vote* + *ws_pos_vote* + *ws_freq_vote*
48:	*l* ← 1 if *total_votes* ≥ 0.5 else 0
49:	*pe* ← calculate the percentage of error per target word in the test data
50:	*ape* ← calculate the average percentage of error for the entire test data
51:	*ne* ← count the number of errors per target word in the test data

The statistical analysis of the four BERT models is performed to provide more insights into word disambiguation. Table 11 shows that the ensemble BERT model's lowest error percentage is reported (3.46 %). Similarly, the maximum number of errors per single target word is 3, equal to the lowest error reported by the single proposed models. Conversely, the baseline model with frequency count as a feature has more errors, as it reported an average error percentage of 7.09 % and a

maximum error value of 6. Including additional features with weak supervision greatly improves the model's performance. It enhances the prediction by increasing the number of words with zero error from 45 to 70 in the frequency count and from 54 to 62 for POS. Similarly, the ensemble BERT model was able to classify the senses for 69 words correctly. Because the ensemble BERT mode has the highest F1 score and the lowest percentage of error, it would be used to compare with the state-of-the-art.

Fig. 12 presents a bar plot for the number of errors for some target words. The target words are chosen to show all cases in predictions. However, the overall performance shows that adding these features improves the model performance, where the total number of errors per target word decreases. The ensemble model has the best performance, where it usually gives the lowest number of errors for a given word.

6.2. Ensemble BERT parameter settings

The ensemble BERT stands out as the best-performing model in WSD. This experiment focuses on conducting a thorough comparative analysis of hyperparameter settings, including batch size and the number of epochs. The objective is to identify and set the optimal hyperparameters to ensure the highest possible model performance.

Table 12 presents the effect of varying batch sizes on the performance of the proposed WSD ensemble BERT model. As the batch size increases from 32 to 256, the F1-score and the number of words with zero errors decrease while the average percentage of error increases. The F1-score is decreasing gradually with the increase in batch size, exhibiting rates of decline of 0.49 %, 1.18 %, and 0.8 % between consecutive batch sizes of 32–64, 64–128, and 128–256, respectively. This degradation in the model performance can be attributed to the increase in batch size in WSD data, resulting in slower updates to the model weights. Consequently, the model tends to memorize the training data rather than generalize. Thus, in WSD, a smaller batch size introduces more frequent updates, injecting stochasticity into the optimization process. This stochasticity is a regularizer, preventing overfitting by discouraging the model from fitting noise in the training data. In the context of WSD, where the dataset is diverse and comprises multiple polysemous words, each possessing various senses, a smaller batch size proves advantageous. A smaller batch size will expose the model to diverse examples, which is beneficial in WSD.

Table 12
Effect of changing batch size on the proposed ensemble BERT model.

Batch Size	F1-score	Average percentage of error (%)	Min error value	Max error value	Number of words with zero error
32	0.95868	3.46	0	3	69
64	0.95397	3.83	0	4	69
128	0.94271	4.46	0	5	61
256	0.93519	5.17	0	4	55

Table 13
Effect of changing the number of epochs on the proposed ensemble BERT model.

Epochs	F1-score	Average percentage of error (%)	Min error value	Max error value	Number of words with zero error
10	0.95868	3.46	0	3	69
20	0.95413	3.53	0	4	69
30	0.96229	3.06	0	4	74

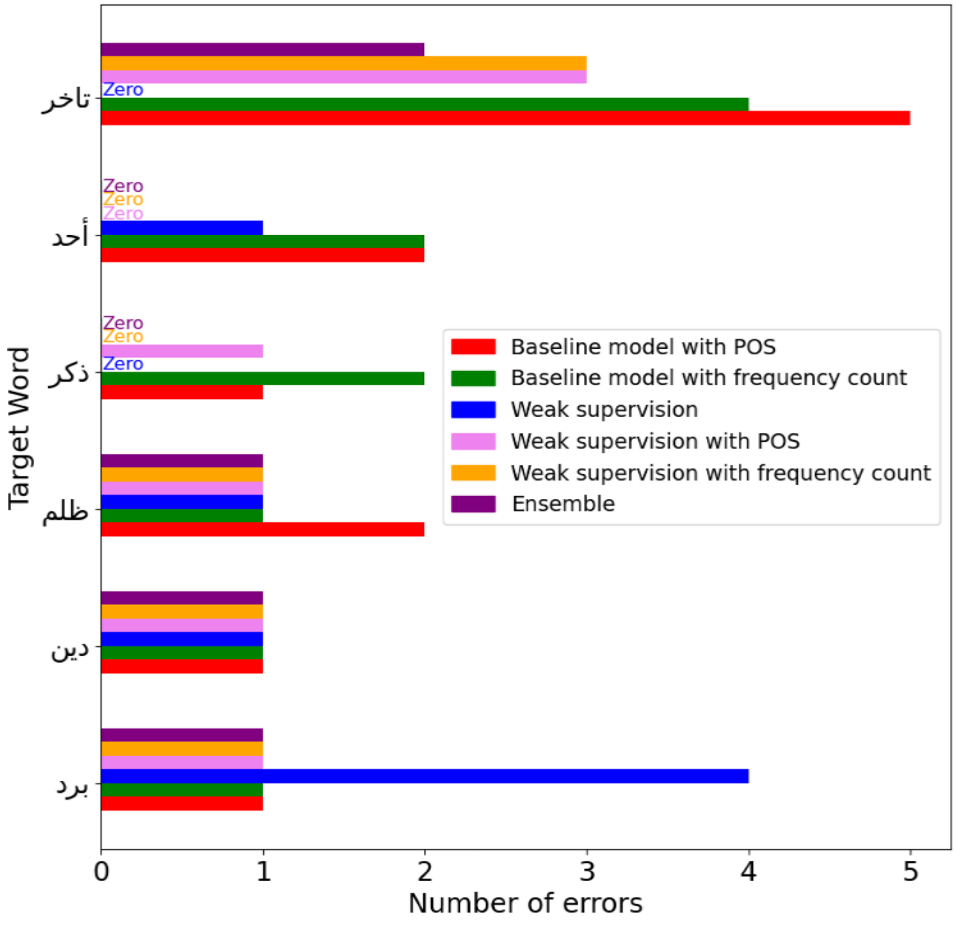

Fig. 12. Number of errors per some target words.

Table 13 presents the performance metrics of the proposed ensemble BERT model across different epochs in terms of F1-score, average percentage of error, minimum error value, maximum error value, and the number of words with zero error. The F1-score achieves its highest value of 0.96229 at 30 epochs. The average error percentage exhibits a slight fluctuation throughout the epochs, with the lowest value of 3.06 % observed at 30 epochs. Table 13 shows a relatively stable performance across the different training durations. The maximum error values increase marginally with the number of epochs, indicating that, although the average error remains low, there are words with higher errors. The number of words with zero errors remains constant at 69 for the first two epochs but increases to 74 at 30 epochs. Thus, as training progresses, the model becomes more adept at correctly classifying more instances with complete accuracy.

The proper selection of hyperparameters is crucial in training an efficient WSD model. The parameters chosen in the proposed WSD model are detailed in Table 14. The following experiments will compare the models with the ensemble BERT model with the parameter settings presented in Table 14. The batch size and the number of epoch values are based on a comprehensive analysis presented in Table 12 and Table 13. The adoption of the Adam optimizer is motivated by its advantageous adaptive learning rate and momentum features. Adam optimizer is one of the most used optimization algorithms in NLP research. Furthermore, the epsilon value is set to prevent division by zero and ensure numerical stability during optimization.

6.3. Comparison with existing WSD pre-trained transformer models in the literature

Since the BERT model has been widely used in WSD, the traditional BERT sequence classification task for English by Huang et al. (2019) was adopted. The WSD BERT model has been implemented on two data shapes (sentence-sense sentence sense with annotated word). The difference between the two data shapes while training BERT is in the signaling target word. The sentence-sense with annotated word data shape is similar to the sentence-sense data shape presented in Table 9, with a single difference that includes the indices of the target word in the sentence. A comparison has been made on the performance of BERT models by adopting Huang et al. (2019) BERT models to the collected and El-Razzaz et al. (2021) Arabic data by utilizing AraBERT Antoun et al. (2020). The choice of adopting Huang et al. (2019) BERT models was due to the availability of the authors' fine-tuned BERT model. Thus, the model hyperparameters are extracted and used for AraBERT Antoun et al. (2020) to build a fair comparison of the algorithm on Arabic WSD datasets.

The algorithm by El-Razzaz et al. (2021) is replicated on their published dataset. This model is also tested on the collected data after reshaping it to match their training data format and compared with them. The reshaping was done by constructing sentence-sense pairs for each sense, where only two samples exist, one true and the other false. A further test is conducted by training their fine-tuned BERT model on the proposed sentence-sense dataset presented in Table 7, where there are more than two samples per sense. This test aimed to demonstrate the impact of data shape in developing a more robust model. The choice of using Arabic Gloss WSD BERT was motivated by the availability of the authors' fine-tuned BERT model and the accompanying dataset. The comparison is based on the model and data found in GitHub El-Razzaz (2021). Thus, a fair comparison could be built to evaluate their algorithm on the generated Arabic WSD dataset in this paper.

Al-Hajj and Jarrar (2022) also proposed another approach that uses the Arabic BERT model with single quotes supervised signal around the target word; however, neither the model nor the data are available. Thus, due to the unavailability of dataset and hyperparameter settings, there is no ability to compare them. They have also conducted a comparative analysis of various pre-trained transformer models, determining that AraBERT stands out as the superior choice. Thus, the CAMelBERT model proposed by Inoue et al. (2021) and the QARiB model developed by Abdelali et al. (2021) have been implemented to validate the effectiveness of the AaBert model on other WSD datasets. These pre-trained models are applied to the generated and El-Razzaz et al. (2021) datasets to evaluate their performance effectively in the context of WSD. Statistical analysis on this dataset is also performed for all adopted models to measure their disambiguation capabilities.

Table 15 compares the proposed ensemble BERT approach and the state-of-the-art pre-trained transformer models on both the collected dataset and the El-Razzaz et al. (2021) dataset. This table presents the F1-score, average, minimum, and maximum error percentage resulting from the models. Table 15 shows that testing the Arabic Gloss WSD El-Razzaz et al. (2021) on the data presented in this paper has resulted in the lowest F1-score of 0.48028. This low score can be justified as only two samples per given word sense. This data distribution leads to a limited learning capability of the model in distinguishing between various senses. The F1-score increased to 0.74646 when utilizing the sentence-sense dataset presented in Table 9, thus incorporating more senses. This F1 score is quite similar to the replicated Arabic gloss WSD proposed by El-Razzaz et al. (2021). However, the performance of these two models on the El-Razzaz et al. (2021) dataset is almost similar. The observed performance can be attributed to the incomprehensive nature of the data.

BERT models proposed by Huang et al. (2019) perform better word disambiguation, resulting in an F1-score of 0.89278 for sentence sense on the collected dataset. The WSD performance is improved to achieve an F1-score of 0.93443 in sentence-sense with annotated words. The BERT-based sentence sense exhibits its lowest performance on the El-Razzaz et al. (2021) dataset, with an F1 score of 0.50903. However, this score increases to 0.84115 when annotating the target word. Annotating the target word increased the model performance because the model can now emphasize the polysemous word.

The other pre-trained models, CAMelBERT and QARiB, have demonstrated effective performance, achieving F1 scores exceeding 0.76 on both datasets. Although several other BERT models have also exhibited strong disambiguation performance, the proposed ensemble BERT model outperforms them, achieving superior results on both datasets. It attains the highest F1 scores, reaching 0.96229 for the collected dataset and 0.85937 for the El-Razzaz et al. (2021) dataset, underscoring its effectiveness in comparison to other models.

Table 15 also presents the average error percentage for all target words. The proposed Ensemble BERT model (3.06 %) on the generated dataset reports the lowest average error percentage. Increasing the number of samples per sense, as in sentence-sense Arabic Gloss BERT, or adding more features, as in sentence-sense with annotated words, has a positive impact by decreasing the average percentage of errors for the collected dataset. Thus, having a more extensive and diverse dataset contributes to better disambiguation results. Moreover, all the methods achieved a minimum error percentage per word of 0 %, implying that they correctly predicted the sense for some words. However, all models, except the ensemble model, proved inadequate in disambiguating certain words, as the models achieved a maximum error percentage of 100 % across the El-Razzaz dataset. The replicated Arabic gloss BERT applied to the collected data also has yielded a 100 % percentage of error for some words in the collected dataset. The high error percentage underscores a substantial limitation in the disambiguation capabilities of

Table 14
Hyperparameter setting.

Parameters	Values
Optimizer	Adam
Learning Rate	2e-5
Epsilon	1e-8
Batch Size	32
Epochs	30

Table 15
Comparison with the BERT models from the literature on the collected dataset.

Tested Models	Collected Dataset				El-Razzaz et al. (2021) Dataset			
	F1 score	**Average percentage of error (%)**	**Min error percentage (%)**	**Max error percentage (%)**	**F1 score**	**Average percentage of error (%)**	**Min error percentage (%)**	**Max error percentage (%)**
Proposed Ensemble BERT	0.96229	3.06	0	44.44	0.85937	15.82	0	50
CAMelBERT Inoue et al. (2021)	0.94061	4.71	0	33.33	0.82736	17.41	0	100
QARiB Abdelali et al. (2021)	0.80972	14.56	0	55.56	0.76184	24.42	0	100
BERT based sentence with annotated word Huang et al. (2019)	0.93443	5.38	0	44.44	0.841156	11.73	0	100
BERT based sentence-sense Huang et al. (2019)	0.89278	9.04	0	44.44	0.509033	17.73	0	100
BERT based sentence-sense Arabic Gloss WSD El-Razzaz et al. (2021)	0.74646	18.36	0	45.45	0.75969	22.75	0	100
Arabic Gloss WSD El-Razzaz et al. (2021)	0.48028	53.35	0	100	0.77885	22	0	100

these models, although an acceptable F1-score of 0.77885 has been achieved. The ensemble model has yielded a superior performance on both datasets, thus effectively addressing and resolving word ambiguity within these specified datasets.

6.4. Comparison with Naïve approach

In the existing literature, several Naïve-Bayes (NB)-based approaches have been proposed by researchers, such as those proposed by Ahmed and Nürnberger (2008), Diab (2004), Elmougy et al. (2008). The first two approaches utilized the NB algorithm for Arabic/English Word Translation Disambiguation, while the latter focused on Arabic WSD. The NB-based WSD approach proposed by Elmougy et al. (2008) is evaluated on the collected dataset presented in this paper to ensure a fair comparison. However, a challenge was encountered as the code implementation by Elmougy et al. (2008) was unavailable to compare the results of their NB-based classifier on different data directly. Thus, their approach was implemented according to the description provided in the paper and tested on the data collected in this paper. Table 16 reported the F1-score of the implementation compared to the proposed Ensemble BERT model, which shows better performance than the NB model. The NB model was not utilized on the El-Razzaz et al. (2021) dataset due to the limitation of having only one example sentence for each sense. This insufficient data hinders the Bayesian network's ability to learn senses and generalize effectively.

Data statistics were conducted to gain insights into the performance of the NB-based classifier. The percentage of error per target word is calculated, which ranges from 0 to 15 %, with an average error rate of 8.88. The average error rate for the proposed Ensemble BERT approach is much lower than that of the NB-based classifier. During these analyses, it was observed that the NB classifier could not predict more than one sense of some words. This indicates the limitations of the NB-based approach. In contrast, the proposed Ensemble BERT could predict a wider range of senses for polysemous words. It is important to note that a direct statistical comparison between the NB-based approach and the approach may not be appropriate due to the inherent differences in data distribution and the number of samples per class. In the NB-based approach, the classes correspond to the senses of each target word. Therefore, the dataset is structured based on the senses of the words, which can result in varying sample sizes for each sense.

Table 16
Comparison of the proposed Ensemble BERT model with Naïve approach.

Approach	**F1-score**	**Average percentage of error (%)**	**Min error value**	**Max error value**
Proposed Ensemble BERT	0.95868	3.46	0	3
Naïve approach Elmougy, Taher, and Noaman (2008)	0.47155	8.8	0	15

7. Case study

The Arabic language poses significant challenges in NLP due to its complexity and lack of a comprehensive lexical database, particularly for polysemous words. Research in Arabic NLP lags, creating a critical need for effective WSD methods to enhance downstream tasks. This case study explores the application of the proposed Ensemble WSD BERT model to improve the performance of sentiment analysis in MSA. This experiment includes four main steps: data collection, data disambiguation, data replacement, and sentiment analysis.

7.1. Data collection

The sentiment analysis dataset in MSA was collected from the web, ensuring each sentence contained an ambiguous word. The dataset is collected so that none of the samples in the sentiment analysis data is presented in the collected WSD data. The dataset, consisting of 1100 samples, was chosen with a specific purpose to test the effectiveness of

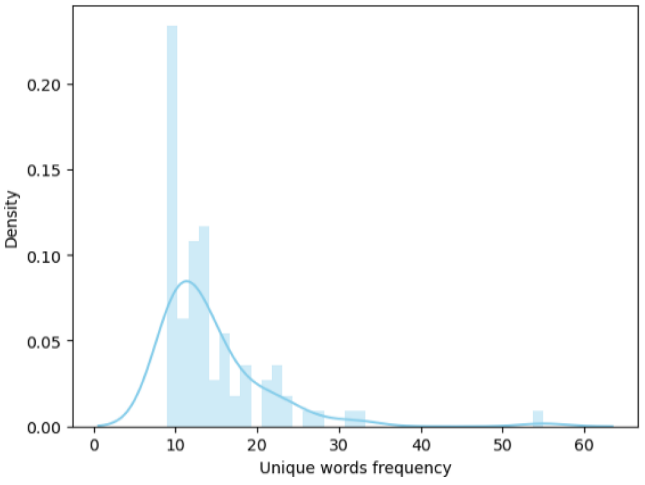

Fig. 14. Distribution of ambiguous words in sentiment analysis data.

WSD in a controlled yet diverse environment. The distribution of sentiment labels was selected to be more balanced than having high variations between labels. It comprises 53 % labeled as positive and 47 % as negative, ensuring that the model does not lean towards a particular sentiment and maintains a fair representation of real-world sentiment distribution.

Figs. 13 and 14 provide valuable insights into the dataset characteristics. Fig. 13 illustrates the average length of sentences, indicating that most sentences range from 5 to 15 words, with the most common length being eight words. This information is essential for understanding the textual context in which sentiment is expressed. Figure 16 delves into ambiguity statistics, revealing the frequency distribution of ambiguous words in the collected sentiment analysis data. The frequency ranges from 9 to 55. This variation of ambiguity reflects the complexities inherent in sentiment analysis, where polysemous words can significantly impact the accuracy of sentiment classification.

7.2. Data disambiguation

The collected data was disambiguated using the proposed Ensemble WSD BERT model. The data were entered as input to the model to predict the sense of each ambiguous word. The model yielded results with the predicted sense of each word. Table 17 presents the F1 score, average percentage of error, and the total number of successfully disambiguated words, where the model could differentiate between different senses. The proposed WSD model on the collected sentiment analysis demonstrates effective generalization on unseen data, achieving an F1 score of 0.9133. The average error percentage was 6.40 %, with 20 correctly classified words without false positives or negatives. This indicates the proposed WSD model generalization ability.

7.3. Data replacement

After the disambiguation process, each polysemous word instance was substituted with its corresponding predicted sense. The output data generated by the WSD model was utilized to replace every ambiguous word with its predicted sense, forming a new set of sentiment analysis data. This step was crucial to guarantee that the sentiment analysis model avoids making erroneous predictions stemming from misinterpreting sentences or inappropriate misunderstandings of words. Consequently, every ambiguous word was systematically replaced by its identified sense. Table 18 exemplifies four samples of sentiment analysis data following this enhancement.

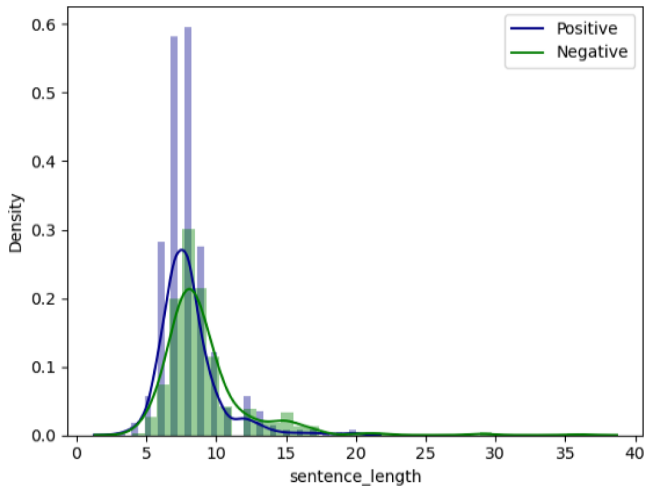

Fig. 13. Sentence length variation per class.

Table 17
Testing results of the proposed Ensemble BERT WSD model on Sentiment Analysis dataset.

Metric	Score
F1 score	0.9133
Average percentage of error	6.40
Number of words with zero errors	20

Table 18
Samples from sentiment analysis data before and after disambiguation.

Original Sentences	Disambiguated Sentences	Sentiment
الإمهال وقلة العناية يؤدي إلى فقدان الحب.	الإمهال وقلة العناية يؤدي إلى فقدان **الحب القمح**.	Negative
الشاب ثار بسبب التمييز والظلم الاجتماعي.	الشاب **غضب غضبا شديدا** بسبب التمييز والظلم الاجتماعي.	Negative
يعتبر التفاني في خدمة الآخرين أحد الفضائل الإنسانية.	يعتبر التفاني في خدمة الآخرين **أعظم**، **أكرم** الفضائل الإنسانية.	Positive
مضى الزائر بعد لحظات جميلة من اللقاء.	**ذهب وابتعد** الزائر بعد لحظات جميلة من اللقاء.	Positive

7.4. Sentiment analysis model

After completing the data replacement process, a Random Forest classifier was applied to the dataset before and after disambiguation. Evaluating the model's performance involved computing performance metrics like precision, recall, and F1 score. Predicted samples were visualized to understand the impact of the disambiguation process on sentiment detection. The dataset was partitioned into training and testing sets with an 80:20 ratio for constructing the Random Forest model.

Table 19 provides insight into the RF model's precision, recall, and F1 score. It includes results when the model was trained on 80 % of the original sentiment data and tested on the remaining 20 %, as well as when it was trained on 80 % of the disambiguated data and tested on the same proportion. Both scenarios used identical samples for training and testing to ensure a fair comparison.

The results in Table 19 show that the Ensemble WSD BERT model demonstrated robust disambiguation performance, significantly improving sentiment analysis results. After disambiguation, the model trained on data reported higher performance metrics (precision, recall, and F1 score). The disambiguated data led to a notable increase in the F1 score, which balances both precision and recall of the sentiment analysis model, from 0.7841 to 0.8072. The increase in F1 score indicates an improvement of 2.31 %.

Fig. 15 provides detailed statistics on the results. The percentage of improvement presented in the figure was evaluated using the formula:

$$\left|\frac{Value_{AfterWSD} - Value_{BeforeWSD}}{Value_{BeforeWSD}}\right| * 100$$

This formula provides a quantitative measure of the percentage of improvement where $Value_{AfterWSD}$ represents the metric's value after the WSD process, and $Value_{BeforeWSD}$ represents the metric's value before the WSD process.

Fig. 15 illustrates a significant improvement in the model's performance, particularly its ability to predict negative samples. The enhancement in handling negative sentiments is remarkable, with a substantial increase of 9.8 %. This improvement underscores the

Table 19
Results of the Random Forest sentiment analysis classifier.

Dataset	Precision	Recall	F1 score
Original Sentiment Analysis Data	0.9237	0.6812	0.7841
Disambiguated Sentiment Analysis Data	0.9406	0.7070	0.8072

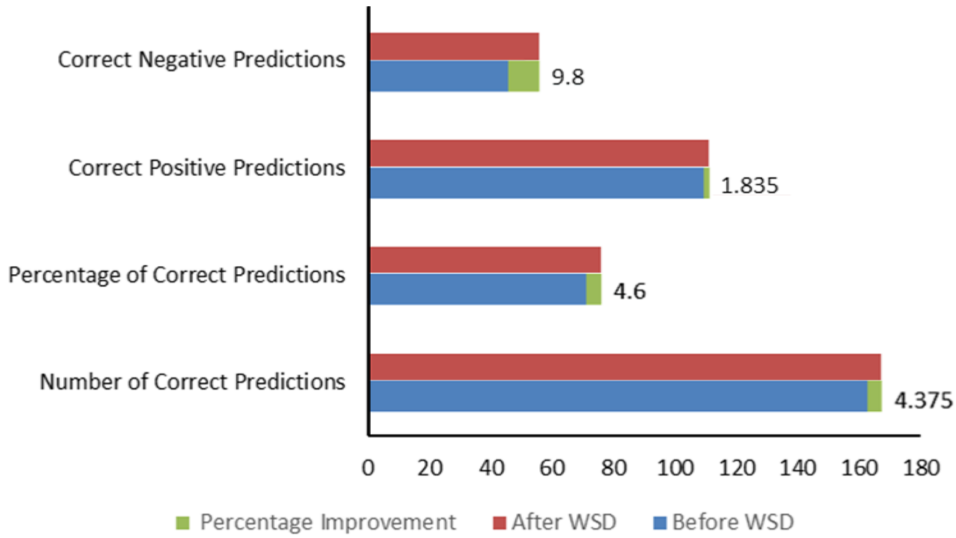

Fig. 15. Sentiment analysis results.

effectiveness of the disambiguation process in addressing challenges that existed before, notably related to the model's bias in associating specific polysemous words with the positive class.

Before disambiguation, the model was predisposed to link certain polysemous words with positive sentiments. This bias was a potential source of inaccuracy in predicting sentiments, especially those leaning toward negativity. The disambiguation process was crucial in mitigating this bias, allowing the model to make more accurate and contextually relevant predictions.

Moreover, the overall correctness of predictions has improved, with a 4.6 % increase. This improvement translates into a 4.375 % increase in the number of correct predictions and a 1.835 % enhancement in positive predictions. Thus, the disambiguation process positively impacted various aspects of the model's predictive capabilities.

Table 20 compares two sentiment prediction samples before and after the WSD process. The sentences, their predicted sentiments before and after WSD, and the actual sentiments are presented in the table. The table shows that sentiment predictions shifted from positive (before WSD) to negative (after WSD) in both sentences. The results in the table indicate that the disambiguation process profoundly impacted the model's understanding, resulting in more accurate sentiment predictions. The change in predictions suggests that polysemous words in the sentences influenced the model's initial sentiment predictions. The disambiguation process likely clarified the intended sense of these words, leading to more contextually accurate sentiment assignments. The first sentence was originally predicted as positive but was later corrected to negative after WSD, highlighting the importance of contextual sensitivity. Disambiguating the meaning of words in context allows the model to discern better the overall sentiment conveyed by the sentence.

Table 21 shows the samples that the model remained unable to predict their polarity even after disambiguation. The model cannot disambiguate the sentiment in the first sentence even after the disambiguation process. Before and after WSD, the model predicted a positive sentiment, while the actual sentiment was negative. This persistent misclassification suggests a continued challenge in capturing the negative implication of the sentence, potentially influenced by the specific complexities or context. In the second sentence, the model maintains the exact sentiment prediction even after WSD. Both predictions are negative, while the real sentiment is positive. Despite the disambiguation attempt, the model's difficulty in recognizing the positive context highlights the complexity of accurately interpreting sentiments in certain linguistic constructs. The persistent misclassifications suggest that the model might face challenges in understanding the contextual cues that influence sentiment. However, although these samples exist but they represent a minority of the predictions.

Table 20
Samples where model predictions are corrected after disambiguation.

Samples	Predicted after WSD	Predicted before WSD	Real label
العمل المتواصل خلال شهر آب يثير شعورا بالإرهاق.	Negative	Positive	Negative
الطلب على المنتجات هوى في الربع الأخير.	Negative	Positive	Negative

Table 21
Samples where model predictions persisted in ambiguity even after disambiguation.

Samples	Predicted after WSD	Predicted before WSD	Real label
لم يمنح له الفرصة للتعبير عن آرائه وأفكاره.	Positive	Positive	Negative
رغم مشاكل الرؤية، يظل لديها عينان تنظران إلى المستقبل.	Negative	Negative	Positive

The obtained sentiment analysis results highlight the effectiveness of the Ensemble WSD BERT model and also emphasize its role in resolving biases and enhancing the model's overall precision. This enhancement contributes to a more reliable and context-aware sentiment prediction, demonstrating the importance of addressing polysemy-related challenges in NLP tasks.

8. Discussion

RQ1 How does the newly created dataset of frequently occurring Arabic words in MSA contribute to the Arabic WSD field?

Answer: The newly created dataset contributes significantly to the field of Arabic WSD. It represents a comprehensive resource for implementing more precise and robust disambiguation algorithms. This dataset offers researchers the advantage of a standardized collection of frequently occurring words. The dataset encompasses highly information-rich content words, including nouns and verbs, as shown in Fig. 5. This diverse corpus provides a high level of contextual information, thereby enhancing the disambiguation process. Moreover, the dataset has been constructed systematically, ensuring each sense is accompanied by ten example sentences. This systematic approach

promotes equal learning among senses, effectively mitigating bias between near senses.

RQ2 How do different data representation techniques impact the suitability of the dataset for Arabic WSD?

Answer: This paper presents a BERT-based WSD model trained using various data representation techniques. The introduction of weak supervision, as detailed in Table 11, complements the existing benchmarks in the field, including sentence sense and sentence with annotated word approaches, as summarized in Table 15. The percentage of error decreases when altering the data configuration. Specifically, different data representation techniques reduce the error percentage from 9.04 % for sentence sense to 5.38 % for sentences with annotated words and further to 4.74 % with the incorporation of weak supervision. This observation shows the importance of annotating the target word with weak supervision, as it significantly enhances the model's proficiency in accurately identifying and classifying polysemous words within the context.

RQ3 How does applying BERT models enhance the disambiguation process?

Answer: BERT consistently delivers impressive results in various NLP tasks, achieving high accuracy with remarkably low error rates. For instance, in WSD, BERT distinguishes between word senses, yielding a high precision rate by accurately identifying the intended meaning of polysemous words. For example, referring to Table 11, the BERT model achieved high results with a low average error percentage. BERT's contextual understanding allows proper disambiguation of a given sense based on the surrounding context, leading to fewer disambiguation errors. The BERT model was able to disambiguate all senses for 70 polysemous words out of 100 when incorporating word frequency count.

RQ4 To what extent does the inclusion of part-of-speech (POS) information improve the accuracy and effectiveness of Arabic WSD models based on BERT?

Answer: The integration of POS tags into BERT-based WSD models plays a significant role in refining their accuracy and effectiveness. POS tags provide vital syntactic cues that assist in disambiguating polysemous words by revealing a word's grammatical role in a sentence. For instance, Table 11 shows an increase in correctly disambiguated words from 54 to 62 when adding POS tags to the BERT with weak supervision. Additionally, Fig. 12 shows a total disambiguation of the word "أحد" (>Hd) in diverse contexts for weak supervision BERT models with POS tag. This shows that POS enables precise differentiation between different tags, thus reducing ambiguity. However, the degree of improvement can vary depending on factors like the quality of POS tagging for the Arabic language.

RQ5 How does considering word frequency affect the performance of Arabic WSD models, mainly when dealing with rare words?

Answer: Word frequency significantly influences the performance of Arabic WSD models, with a pronounced effect when dealing with rare words. This is shown in Table 11, where the model of weak supervision with frequency counts has the highest number of totally disambiguated words. This means that it was able to disambiguate rare words where the previous BERT model failed to disambiguate. In addition, Zipf's principle, introduced in 1945 Zipf (1945), establishes a fundamental connection between word frequency and the number of meanings. According to this principle, words occurring more frequently in a language tend to have a more significant number of meanings. Arabic WSD models capitalize on this principle to guide their disambiguation process. When dealing with rare words, the proposed models prioritize contextual information to compensate for the scarcity of frequency-driven sense distinctions. For instance, referring to Fig. 12, a less common term such as "برد" (barod), the percentage of error decreased from 4 % to 1 % by adding frequency count to weak supervision.

RQ6 What benefits does the weighted ensemble approach offer in Arabic WSD, and how does it contribute to the disambiguation of specific word senses?

Answer: The weighted ensemble approach presents several benefits in the Arabic WSD domain, significantly enhancing the disambiguation of specific word senses. This approach can combine multiple models' outputs, thereby capitalizing on the strengths of individual models and mitigating their weaknesses. The weighted ensemble approach achieves the highest F1 score, as in Table 11. Simultaneously, it has the lowest percentage of errors, proving its ability to minimize disambiguation errors. This approach takes advantage of incorporating POS tags, word frequency, and weak supervision, thus disambiguating both frequent and rare terms.

RQ7 Under what conditions and to what extent do BERT-based approaches outperform traditional methods, like Naive Bayes, in Arabic WSD?

Answer: BERT-based approaches excel over traditional methods, such as Naive Bayes in Arabic WSDs. This is shown from the results presented in Table 13, with BERT-based approaches yielding a notably high F1-score of 0.95868 and a remarkably low average error rate of 3.46 %. This showcases their ability to balance precision and recall, translating into heightened accuracy. Furthermore, the consistent superiority of BERT-based approaches across the F1-score, error rate metrics, and the range of error values illustrates their reliability and robustness. Their capacity to outperform traditional methods extends to a broad spectrum of word senses and contexts, making them the preferred choice for accurate and versatile Arabic WSD applications and emphasizing their indispensable role in advancing the field.

RQ8 What are the predominant errors the proposed Arabic WSD system makes, and do they reveal common patterns or linguistic challenges that must be addressed?

Answer: The analysis of the error types made by the proposed Arabic WSD system offers crucial insights into common linguistic challenges and patterns. Referring to Fig. 12, disambiguating some words like "دين" (dyn) has a consistent output regardless of the model, features, and data representation technique used. This analysis underscores the need for addressing domain-specific language usages, necessitating tailored strategies for improvement.

RQ9 How does the scarcity of Arabic language resources, such as sense-annotated corpora and lexicons, affect the performance and feasibility of Arabic WSD systems?

Answer: The scarcity of Arabic language resources, including sense-annotated corpora and lexicons, impacts the performance and feasibility of Arabic WSD systems. Insufficient resources hinder the development and training of robust WSD models, as they rely on large-scale, high-quality data for effective performance. Table 10 shows that the replicated BERT model, which is trained on the El-Razzaz *et al.* dataset, has a low F1 score and failed to disambiguate some polysemous words. The word disambiguation failure happened due to the non-comprehensiveness and low diversity of the dataset, although it is considered large data. Additionally, limited sense-annotated corpora make the generalization of WSD models challenging.

9. Conclusion and future work

The ambiguity of Arabic words increases for several reasons, like missing diacritics and normalized data in most published articles. Arabic is also agglutinative and complex, containing similar words with different senses at multiple levels. This paper presents disambiguating Arabic words using BERT. It incorporates new data features that strongly correlate with target polysemous words. Ensemble BERT was introduced to get an enhanced WSD BERT model. Weighted voting, which gives higher weights to better models, was used to build the ensemble model. The statistics showed that incorporating these features improves the model performance, although the F1-score slightly increased. A sense extensive dataset is collected to perform the task of word disambiguation. The proposed approach has outperformed other adopted benchmark algorithms. The proposed Ensemble WSD model exhibits robust generalization on unseen data, extending its effectiveness to enhance downstream tasks such as sentiment analysis.

In the future, the work will involve expanding the dataset used for

WSD and re-running the experiments on a larger and more diverse data corpus. This expansion could significantly enhance the model's performance, exposing it to more diverse language contexts and complexities and enabling a more robust WSD. The increased data volume could lead to improved model generalization and effectiveness in handling more polysemous words and their senses, making it a valuable direction for further research in Arabic NLP, like machine translation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is funded by Zayed University Research Incentive Fund (RIF) with grant number [R22047].

References

Abdelaali, B., Tlili-Guiassa, Y., 2022. Swarm optimization for Arabic word sense disambiguation based on English pre-trained word embeddings, pp. 1–6. 10.1109/ISIA55826.2022.9993494.
Abdelali, A., Hassan, S., Mubarak, H., Darwish, K., & Samih, Y. (2021). Pre-training BERT on Arabic tweets: Practical considerations. arXiv preprint arXiv:2102.10684.
Abderrahim, M.A., Mohammed El Amine, A., 2022. Arabic word sense disambiguation for information retrieval. ACM Transactions on Asian and Low-Resource Language Information Processing 21, 1–19. 10.1145/ 3510451.
Abou Khalil, V., Helou, S., Flanagan, B., Chen, M.R., Ogata, H., 2019. Learning isolated polysemous words: identifying the intended meaning of language learners in informal ubiquitous language learning environments. Smart Learning Environments 6. https://doi.org/10.1186/s40561-019-0095-0.
Abu El-Khair, I., 2016. 1.5 billion words Arabic corpus. arXiv preprint arXiv:1611.04033. 10.48550/arXiv.1611.04033.
Ahmed, F., Nürnberger, A., 2008. Arabic/English word translation disambiguation using parallel corpora and matching schemes, in. In: Proceedings of the 12th Annual Conference of the European Association for Machine Translation, pp. 6–11.
Albared, M., Omar, N., Ab Aziz, M., 2009. Classifiers combination to Arabic morphosyntactic disambiguation, pp. 163–171. 10.1109/ICEEI.2009.5254797.
Al-Hajj, M., Jarrar, M., 2022. ArabGlossBERT: Fine-tuning BERT on contextgloss pairs for wsd. arXiv preprint arXiv:2205.09685. 10.48550/arXiv.2205.09685.
Alian, M., Awajan, A., 2020, November. Sense inventories for arabic texts. IEEE, pp. 1–4. https://doi.org/10.1109/ACIT50332.2020.9300054.
Alian, M., Awajan, A., 2023. Arabic word sense disambiguation using sense inventories. Int. J. Inf. Technol. 15 (2), 735–744.
Alian, M., Awajan, A.A., Al-Kouz, A.-K., 2016. Arabic Word Sense Disambiguation Using Wikipedia. IJCIS 12 (1), 61–66.
Alian, M., Awajan, A., Al-Hasan, A., Akuzhia, R., 2019. Towards building Arabic paraphrasing benchmark, pp. 1–5. 10.1145/3368691.3368708.
AlJazeera, 2023a. Aljazeera documentary: Stay updated on what's happening around the world. Al Jazeera Media Network. https://doc.aljazeera.net/.
AlJazeera, 2023b. Aljazeera net: Latest news of the day from around the world. Al Jazeera Media Network. https://www.aljazeera.net/.
Alkhatlan, A., Kalita, J., Alhaddad, A., 2018. Word sense disambiguation for Arabic exploiting Arabic wordnet and word embedding. Procedia Comput. Sci. 142, 50–60. https://doi.org/10.1016/j.procs.2018.10.460.
Almaany, 2023. Multilingual and multidisciplinary dictionary of meanings - arabic-arabic dictionary. https://www.almaany.com/.
Al-Maghasbeh, M., Hamzah, M., 2015. Extract the semantic meaning of prepositions at Arabic texts: An exploratory study. Int. J. Computer Trends Technol. 30, 116–120. https://doi.org/10.14445/22312803/IJCTT-V30P121.
AlMousa, M., Benlamri, R., Khoury, R., 2022. A novel word sense disambiguation approach using wordnet knowledge graph. Comput. Speech Lang. 74, 101337 https://doi.org/10.1016/j.csl.2021.101337.
Alqahtani, S., Aldarmaki, H., Diab, M., 2019. Homograph disambiguation through selective diacritic restoration.
Alsharekh, M., 2019. The Contemporary Dictionary. https://lexicon.alsharekh.org/.
Al-sulaiti, L., Atwell, E., 2003. The design of a corpus of contemporary Arabic. 10.13140/2.1.2228.8320.
Altibbi, 2023. Altibbi website for health information and medical consultations: Diseases, medications, and treatment. Altibbi FZ-LLC. https://altibbi.com/.
Antoun, W., Baly, F., Hajj, H., 2020. AraBERT: Transformer-based model for Arabic language understanding. arXiv preprint arXiv:2003.00104 10.48550/arXiv.2003.00104.
ArabiaWeather, 2023. Arabiaweather: Weather news & forecast for today and tomorrow. ArabiaWeather, Inc. https://www.arabiaweather.com/.
ArabicPost, 2023. ArabicPost. Integral Media Danismanlik Şti Limited or its licensors. https://arabicpost.net/.
Argaam, 2023. Argaam: News and information about the Saudi stock market - tadawul. Argaam Investment. https://www.argaam.com/.
Banerjee, S., Pedersen, T., 2003. Extended gloss overlaps as a measure of semantic relatedness, in: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence - IJCAI-03, pp. 805–810.
BBC, 2023. BBC News Arabic - homepage. https://www.bbc.com/arabic.
Belinkov, Y., Magidow, A., Romanov, M., Shmidman, A., Koppel, M., 2016. Shamela: A large-scale historical Arabic corpus. arXiv preprint arXiv:1612.08989. 10.48550/arXiv.1612.08989.
Boudabous, M.M., Belguith, L.H., Sadat, F., 2013. Exploiting the Arabic wikipedia for semi-automatic construction of a lexical ontology. Int. J. Metadata Semant. Ontol. 8, 245–253. https://doi.org/10.1504/IJMSO.2013.057768.
Bouhriz, N., Benabbou, F., Benlahmar, E.H., 2016. Word sense disambiguation approach for Arabic text. International Journal of Advanced Computer Science and Applications 7. 10.14569/IJACSA.2016.070451.
CNN, 2023. CNN Arabic - latest political, sports, and entertainment news and video reports. Cable News Network. https://arabic.cnn.com/.
Corrêa Jr, E.A., Amancio, D.R., 2019. Word sense induction using word embeddings and community detection in complex networks. Phys. A 523, 180–190. https://doi.org/10.1016/j.physa.2019.02.032.
Corrêa Jr, E.A., Lopes, A.A., Amancio, D.R., 2018. Word sense disambiguation: A complex network approach. Inf. Sci. 442, 103–113. https://doi.org/10.1016/j.ins.2018.02.047.
Debili, F., Achour, H., Souissi, E., 2002. La langue Arabe et l'ordinateur: de l'étiquetage grammatical à la voyellation automatique. Correspondances 71, 10–28.
Diab, M., Resnik, P., 2002. An unsupervised method for word sense tagging using parallel corpora, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 255–262. 10.3115/1073083.1073126.
Diab, M., 2004. An unsupervised approach for bootstrapping Arabic sense tagging, in: Proceedings of the workshop on computational approaches to Arabic script-based languages, pp. 43–50. 10.3115/1621804.1621818.
Djaidri, A., Aliane, H., Azzoune, H., 2023. The contribution of selected linguistic markers for unsupervised Arabic verb sense disambiguation. ACM Trans. Asian Low-Resour. Lang. Inf. Process. 22. URL: https://doi.org/10.1145/3605777.
El-Gedawy, M.N., 2013. Using fuzzifiers to solve word sense ambiguation in Arabic language. International Journal of Computer Applications 79, 1–8.
Elmougy, S., Hamza, T., Noaman, H., 2008. Naïve Bayes classifier for Arabic word sense disambiguation. In: In: Proceeding of the 6th International Conference on Informatics and Systems, pp. 16–21.
El-Razzaz, M., 2021. Arabic word sense disambiguation benchmark. Arabic-word-sense-disambiguation-bench-mark.
El-Razzaz, M., Fakhr, M., Maghraby, F., 2021. Arabic gloss WSD using bert. Appl. Sci. 11, 2567. https://doi.org/10.3390/app11062567.
Farghaly, A., Farghaly, A., Shaalan, K., Khaled, 2009. Arabic natural language processing: Challenges and solutions. ACM Transactions on Asian Language Information Processing (TALIP) 8.
Fellbaum, C., Alkhalifa, M., Black, W., Elkateb, S., Pease, A., Rodriguez, H., Vossen, P., 2006. Introducing the Arabic wordnet project, in: Sojka, P., Choi, K.S., Fellbaum, C., Vossen, P. (Eds.), Proceedings of the 3rd Global Wordnet Conference, Jeju Island, Korea, South Jeju, January 22-26, 2006. Proceedings of the 3rd Global Wordnet Conference.
Fouad, M.M., Mahany, A., Aljohani, N., Abbasi, R.A., Hassan, S.-U., 2020. Arwordvec: efficient word embedding models for Arabic tweets. Soft. Comput. 24 (11), 8061–8068.
Foundation, W., 2023. Wikipedia the free encyclopedia. https://ar.wikipedia.org/wiki/.
Gonzalo, J., Chugur, I., Verdejo, F., 2000. Sense clusters for information retrieval: Evidence from semcor and the EuroWordNet InterLingual index, in: ACL-2000 Workshop on Word Senses and Multi-linguality, Association for Computational Linguistics, Hong Kong, China. pp. 10–18. 10.3115/1117724.1117726.
Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T., 2018. Learning word vectors for 157 languages, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), Miyazaki, Japan. URL: https://aclanthology.org/L18-1550.
Habib, M., Faris, M., Alomari, A., Faris, H., 2021. Altibbivec: A word embedding model for medical and health applications in Arabic language. IEEE Access PP, 1–1. 10.1109/ACCESS.2021.3115617.
Huang, L., Sun, C., Qiu, X., Huang, X., 2019. GlossBERT: Bert for word sense disambiguation with gloss knowledge. arXiv preprint arXiv:1908.07245. 10.48550/arXiv.1908.07245.
Ide, N., Véronis, J., 1998. Word sense disambiguation: The state of the art. Comput. Linguist. 24, 1–41.
Inoue, G., Alhafni, B., Baimukan, N., Bouamor, H., & Habash, N. (2021). The interplay of variant, size, and task type in Arabic pre-trained language models. arXiv preprint arXiv:2103.06678.
Kaddoura, S., D. Ahmed, R., D., J.H., 2022. A comprehensive review on Arabic word sense disambiguation for natural language processing applications. WIREs Data Mining and Knowledge Discovery 12. 10.1002/widm.1447.
Kaddoura, S., Itani, M., Roast, C., 2021. Analyzing the effect of negation in sentiment polarity of Facebook dialectal Arabic text. Appl. Sci. 11, 4768. https://doi.org/10.3390/app11114768.
Kaddoura, S., Alex, S.A., Itani, M., Henno, S., AlNashash, A., Hemanth, D.J., 2023. Arabic spam tweets classification using deep learning. Neural Comput. & Applic. 35 (23), 17233–17246.
Kilgarriff, A., Yallop, C., 2000. What's in a thesaurus?, in: Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00),

European Language Resources Association (ELRA), Athens, Greece. URL: http://www.lrec-conf.org/proceedings/lrec2000/pdf/180.pdf.

Laatar, R., Chafik, A., Belguith, L., 2018. Word embedding for Arabic word sense disambiguation to create a historical dictionary for Arabic language, pp. 131–135. 10.1109/CSIT.2018.8486159.

Lu, W., Meng, F., Wang, S., Zhang, G., Zhang, X., Ouyang, A. and Zhang, X., 2019. Graph-Based Chinese Word Sense Disambiguation with Multi-Knowledge Integration. Computers, Materials & Continua, 61(1). 10.32604/cmc.2019.06068.

Merhbene, L., Zouaghi, A., Zrigui, M., 2013. A semi-supervised method for Arabic word sense disambiguation using a weighted directed graph, in: Proceedings of the Sixth International Joint Conference on Natural Language Processing, Asian Federation of Natural Language Processing, Nagoya, Japan. pp. 1027–1031. URL: https://aclanthology.org/I13-1140.

Merhbene, L., Zouaghi, A., Zrigui, M., 2014. An approach based on semantic trees for lexical disambiguation of Arabic language using a voting procedure, in: TALN-RECITAL 2014 Workshop RLTLN 2014: Réseaux Lexicaux pour le TAL (RLTLN 2014: Lexical Networks for NLP), Association pour le Traitement Automatique des Langues, Marseille, France. pp. 281–290. URL: https://aclanthology.org/W14-6702.

Miller, G.A., 1992. Wordnet: A lexical database for English, in: Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992. URL: https://aclanthology.org/H92-1116.

Pedersen, T., 2007. Learning probabilistic models of word sense disambiguation. arXiv preprint arXiv:0707.3972. 10.48550/arXiv.0707.3972.

Quispe, L.V., Tohalino, J.A., Amancio, D.R., 2021. Using virtual edges to improve the discriminability of co-occurrence text networks. Phys. A 562, 125344. https://doi.org/10.1016/j.physa.2020.125344.

Saidi, R., Jarray, F., Alsuhaibani, M., 2022. Comparative analysis of recurrent neural network architectures for Arabic word sense disambiguation, pp. 272–277. 10.5220/0011527600003318.

Saidi, R., Jarray, F., Akacha, A. and Aribi, W., 2023, September. WSDTN a Novel Dataset for Arabic Word Sense Disambiguation. In International Conference on Computational Collective Intelligence (pp. 203-212). Cham: Springer Nature Switzerland. 10.1007/978-3-031-41774-0_16.

Saif, A., Omar, N., Zainodin, U.Z., Ab Aziz, M.J., 2018. Building sense tagged corpus using Wikipedia for supervised word sense disambiguation. Procedia Comput. Sci. 123, 403–412. https://doi.org/10.1016/j.procs.2018.01.062.

Shifa, 2023. Shifaa: A 24/7 renewed medical platform. shifaa platform. https://www.shifaa.ma/.

Taghipour, K., Ng, H., 2015. Semi-supervised word sense disambiguation using word embeddings in general and specific domains, pp. 314–323. 10.3115/v1/N15-1035.

Yarowsky, D., 1995. Unsupervised word sense disambiguation rivaling supervised methods, in. Annual Meeting of the Association for Computational Linguistics.

Zipf, G.K., 1945. The meaning-frequency relationship of words. J. Gen. Psychol. 33 (2), 251–256.

Zouaghi, A., Merhbene, L., Zrigui, M., 2011. Word sense disambiguation for Arabic language using the variants of the Lesk algorithm. WORLDCOMP 11, 561–567.

Zouaghi, A., Merhbene, L., Zrigui, M., 2012a. Combination of information retrieval methods with Lesk algorithm for Arabic word sense disambiguation. Artif. Intell. Rev. 38 (4), 257–269.

Zouaghi, A., Merhbene, L., Zrigui, M., 2012b. Zouaghi, anis and merhben, laroussi and zrigui, mounir. International Journal of Computer Processing of Languages 24, 133–152.