ASOS at KSAA-CAD 2024: One Embedding is All You Need for Your Dictionary

Serry Sibaee[1], Abdullah I. Alharbi[2], Samar Ahmed, Omer Nacar[1], Lahouri Ghouti[1], Anis Koubaa[1]

[1]Robotics and Internet-of-Things Lab, Prince Sultan University, Riyadh 12435, Saudi Arabia
[2]Faculty of Computing and Information Technology in Rabigh
King Abdulaziz University, Jeddah, Saudi Arabia
{ssibaee,onajar,lghouti, akoubaa}@psu.edu.sa, Samar.sass6@gmail.com, aamalharbe@kau.edu.sa

Abstract

Semantic search tasks have grown extremely fast following the advancements in large language models, including the Reverse Dictionary and Word Sense Disambiguation in Arabic. This paper describes our participation in the Contemporary Arabic Dictionary Shared Task. We propose two models that achieved first place in both tasks. We conducted comprehensive experiments on the latest five multilingual sentence transformers and the Arabic BERT model for semantic embedding extraction. We achieved a ranking score of 0.06 for the reverse dictionary task, which is double than last year's winner. We had an accuracy score of 0.268 for the Word Sense Disambiguation task.

1 Introduction

A Reverse Dictionary (RD) takes a phrase that describes a specific concept as input and provides words whose definitions match the entered phrase. This is different from the traditional function of a forward dictionary, which maps words to their meanings or definitions, helping users understand unfamiliar terms. For example, a forward dictionary would tell the user that (ألبد) means (ركب الشيء بعضه على بعض - things are superimposed on top of each other) whereas an RD allows the user to input the phrase (ركب الشيء بعضه على بعض - things are superimposed on top of each other) and would likely produce the word (ألبد) along with other words having similar meanings as the output. While numerous RDs exist for languages like English, Japanese, and French, Arabic has seen limited development in this area, often due to the absence of substantial datasets with word definitions.

RDs are extremely helpful in overcoming the tip-of-the-tongue phenomenon, where people struggle to find the right word to express their thoughts. Additionally, RDs are essential in applications of NLP such as evaluating sentence representations, mapping text to specific entities, answering questions, and retrieving information. It is crucial to understand the precise meanings of words in context. This understanding is improved by Word Sense Disambiguation (WSD), which helps to identify the most relevant meaning of a word in a given context, thus enhancing the effectiveness and accuracy of reverse dictionaries in handling complex queries. For example, the phrase "to come together" could correspond to various options like "meet," "gather," "assemble," and more. Each of these words, while synonyms, may be more appropriate in different contexts, highlighting the importance of WSD. WSD plays a significant role in improving the accuracy of RD systems by helping to determine which word meanings are most relevant based on contextual cues.

This paper describes our participation in the Contemporary Arabic Dictionary Shared Task, organized by the King Salman Global Academy for Arabic Language (KSAA-CAD). The task aims to tackle challenges in the Arabic linguistic domain, focusing on two main areas: RD and WSD. In this study, we extend the methodologies previously developed in our work (Sibaee et al., 2023), utilizing an expanded model further to enhance the performance and capabilities of our approach. By participating in both tasks, we aim to contribute valuable insights and advancements to the fields of RD and WSD, particularly within the context of the Arabic language, where such resources have historically been scarce.

2 Related Work

Previously, researchers have tackled the RD problem using a traditional approach called semantic analysis, utilizing WordNet (Méndez et al., 2013). They employed semantic similarity measurements to determine how similar two words are, using

Proceedings of The Second Arabic Natural Language Processing Conference, pages 697–703
August 16, 2024 ©2024 Association for Computational Linguistics

distance-based similarity measures to establish connections between the term and the input words in a graph (Thorat and Choudhari, 2016). Recently, many researchers have been utilizing embedding techniques alongside advanced neural networks to enhance the generation of reverse dictionaries. Pilehvar (2019) implemented a combination of Bidirectional Long Short-Term Memory (BiLSTM) and Cascade Forward Neural Network (CFNN) to improve the performance of neural RDs. To assess whether a proposed neural network framework is universally effective across all languages, Bendahman et al. (2022) employed sequential models with a variety of neural networks, including embedding networks, denser networks, and Long Short-Term Memory (LSTM) networks.

For Arabic RD, Al-Matham et al. (2023) proposed the first shared task focused on developing RD systems for the Arabic language. This encompassed two subtasks: Arabic RD and Cross-lingual RD. The aim was to enable users to locate words based on their meanings or definitions in both Arabic and English. Sibaee et al. (2023), proposed a method that converts word definitions into multidimensional vectors and trains them using the SemiDecoder model. The proposed system secured the 2nd place based on the rank metric for target embeddings (Electra and SGNS). Elbakry et al. (2023) proposed a method utilizing an ensemble of fine-tuned Arabic BERT-based models to predict the word embedding for a given definition. The final representation is obtained by averaging the output embeddings from each model within the ensemble.

In studies on the WSD task, Barba et al. (2021) introduced CONtinuous SEnse Comprehension (CONSEC), a novel approach designed to address limitations in traditional supervised neural WSD systems. In conventional systems, each word is disambiguated independently without considering the explicit senses assigned to nearby words. Focusing on eight languages, CONSEC introduced a feedback loop strategy that conditions the disambiguation process on both the context and possible meanings of a word, as well as the senses assigned to adjacent words. Kaddoura and Nassar (2024) proposed the development of a dataset for Arabic WSD, utilizing an ensemble learning architecture that combined multiple BERT models. The ensemble model aimed to address word ambiguity effectively and enhance classification performance by integrating various models, each equipped with unique feature sets. These feature sets included part-of-speech (POS) information, word frequency counts, and weak supervision annotations, which contributed valuable contextual information to the disambiguation process. Saidi and Jarray (2023) tackled the task of Word Sense Induction (WSI), a similar NLP task that plays a pivotal role in understanding and interpreting the nuanced meanings of language. Their proposed approach for Arabic WSI seeked to group sentences based on their semantic similarity. This approach involved a two-stage clustering method. In the first stage, sentences were encoded using BERT or DistilBERT to generate sentence embeddings. In the second stage, clustering algorithms like K-means and hierarchical agglomerative clustering were used to partition sentences based on their semantic similarity. The model specifically employed the K-means algorithm to cluster the sentence embedding vectors.

3 Methodology

In this section we will describe the used methodology that was used in the paper by starting of by the description of the dataset and our approach to expand subset of it, Then the explanation about the used model and how we expanded our approach in our past paper (Sibaee et al., 2023), and for the second task WSD we will mention it in end of each section.

3.1 Data Description

3.1.1 RD Task

This year, the task dataset was provided as follows: 31,372 samples for training, 3,921 samples for development, and 3,922 samples for testing. Each training and development data sample includes the definition (gloss) and the targeted word in both text and embedding formats. Moreover, the given embeddings were from three different models: Electra (256-d) (Antoun et al., 2021), camelBERT-MSA (768-d) (Inoue et al., 2021), and AraBERTv2 (768-d) (Antoun et al., 2020). We expanded the dataset using the previous year's dataset and only included the ArElectra embeddings, which were provided last year and comprised approximately 84000 samples, including training and development data. Due to the limited data provided this year, we combined the training and development datasets. This decision was based on our analysis and insights from the previous year, which will be discussed in the following subsection. For this reason, we will not report results based on the development subset.

3.1.2 WSD Task

The WSD dataset is provided in JSON format and consists of two main components: the WSD context gloss mapping and the dictionary data. The WSD context gloss mapping includes elements such as word, context (the sentence or phrase providing semantic clues), context ID, and corresponding gloss ID. The dictionary data comprises word, gloss (the definition of the word aiding in disambiguation), and gloss ID. The dataset is distributed across three subsets: 22,404 entries for training, 2,801 for development, and 2,801 for testing, with a total of 15,865 gloss entries in the WSD dictionary.

3.2 Models

In previous work (Sibaee et al., 2023), our approach centred on utilizing a consistent sentence transformer model while exploring various neural network architectures. In contrast, the current study adopts an inverse strategy, where we standardized the neural network architecture. Firstly, we will explain the neural network architecture for both tasks and then discuss the sentence transformer models used in our experiments.

3.2.1 Neural Network Architecture for RD

After conducting extensive experiments, we optimized our neural network architecture by selecting a semi-encoder structure with four hidden layers. These layers decrease in size by a factor of two, starting from eight times the input size down to the input size itself (input=s, h1=s*8, ..., h4=s, output). We used GELU activation and adjustable dropout rates ranging from 0.2 to 0.4. This configuration was standardized for all experiments, with the only modification being the optimizer, which was changed to AdamW. Additionally, we adjusted the learning rate from 1e-3, used in our previous work, to 1e-4 after finding the former to be too high for this architecture. We maintained this learning rate across all experiments. For abstract architecture please look at figure 1.

3.2.2 Neural Network Architecture for WSD

In this task, we carefully designed a two-stage neural network architecture. Initially, each input is processed independently (where the targeted word and the context are both embedded using ArBERTv2 model) through its dedicated neural network layer,

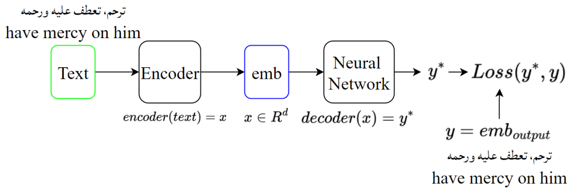

Figure 1: abstract architecture of the Reverse dictionary pipeline

consisting of an input layer, a single hidden layer, and an output layer. The outputs from these individual networks are then concatenated and fed into a subsequent neural network that mirrors the structure of the first, including an input layer, a hidden layer, and an output layer (which has the real output embedding using the ArBERTv2 model also). The final output is a precisely defined 768-dimensional vector. Detailed discussion on the input encoding will be presented in the upcoming section, providing deeper insight into the methodology employed. Worth mentioning: the texts are embedded using ArBERTv2 model in the section 3.2.3 in ArBERTv2 paragraph.

3.2.3 Encoding models for the both tasks

Instead of only using the 'distiluse-base-multilingual-cased-v1' model (Reimers and Gurevych, 2019) to encode texts, the creators of this model have recently developed additional sentence transformer models, now available on the Hugging Face platform. We conducted a comprehensive evaluation of these models to assess their performance enhancements; details of these models are in Table 1.

Sentence transformers model	Dimension
LaBSE	768
MiniLM-L12-v2	384
mpnet-base-v2	768
cased-v1	512
cased-v2	512

Table 1: Embedding Models

We employed various sentence transformer models in our experiments. The LaBSE model (Feng et al., 2022), a language-agnostic BERT sentence embedding model, encodes text into high-dimensional vectors. This model is trained and optimized to produce similar representations exclusively for bilingual sentence pairs that are translations of each other, and it supports 109 different languages.

The MiniLM-L12-v2 model[1] is another sentence transformer that maps sentences and paragraphs to a 384-dimensional dense vector space, making it suitable for tasks such as clustering and semantic search. Similarly, the mpnet-base-v2 model[2] maps sentences and paragraphs to a 768-dimensional dense vector space and can be used for similar tasks.

The Distiluse-base-multilingual-cased models[3], both v1 and v2, map sentences and paragraphs to a 512-dimensional dense vector space. We specifically used the v1 model because it was utilized in our previous research.

Additionally, in our final experiments, we incorporated ArBERTv2 (Abdul-Mageed et al., 2021), (Elmadany et al., 2022), a BERT model (Devlin et al., 2019) trained on approximately 250 GB of Modern Standard Arabic (MSA) text. To represent an entire sentence, we calculated the mean of the token embeddings produced by the model:

$$x = \frac{1}{N} \sum_{1}^{N} e_i$$

where x is the input in the neural network and $e_i \in R^{768}$ is the embedding of each token from the ArBERTv2 model, Also for the WSD task we used the same model to represent the context, targeted word, and the output from the dictionary.

4 Result

In this section we will describe our results from training the model mentioned in section 3.2.1 but before that we will mention an experiment that we did while working on the task.

4.1 RD results

4.1.1 Sentence Transformers Results

This section presents the results of the sentence transformer encoders method, focusing on the RANK metric, as shown in Table 2. For the Electra model, LaBSE achieved the best RANK score (0.17), followed by mpnet and L12 with scores of 0.22, and CasedV2 and CasedV1 scoring 0.25 and 0.21, respectively. The bertMSA model demonstrated improved RANK scores, with CasedV2 leading at 0.403, followed by mpnet (0.371), L12 (0.368), and LaBSE (0.367). In the BERTseg model, CasedV2 again achieved the highest RANK score (0.416), with mpnet and LaBSE both scoring 0.399, and L12 and CasedV1 scoring 0.397 and 0.406, respectively. Overall, CasedV2 and LaBSE consistently performed well on the RANK metric across all models, demonstrating their effectiveness in this task.

Encoder	RANK	COS	MSE
Electra			
mpnet	0.22	0.562	0.225
L12	0.22	0.56	0.22
LaBSE	0.17	0.57	0.221
CasedV2	0.25	0.55	0.23
CasedV1	0.21	0.56	0.226
bertMSA			
mpnet	0.371	0.717	0.327
L12	0.368	0.718	0.326
LaBSE	0.367	0.719	0.324
Cased V2	0.403	0.712	0.331
Cased V1	0.378	0.716	0.327
BERTseg			
mpnet	0.399	0.778	0.076
L12	0.397	0.778	0.076
LaBSE	0.399	0.780	0.0755
CasedV2	0.416	0.777	0.076
CasedV1	0.406	0.779	0.075

Table 2: Results of the sentence transformer models (1800 epochs, 0.2 dropout) on the test dataset

4.1.2 ArBERTv2 Results

This section presents the best results from our final proposed method using the ArBERTv2 embedding, as displayed in Table 3. The Electra model achieved the highest RANK score (0.0644) after 500 epochs. Following closely, the bertMSA model attained a RANK score of 0.1484 after 1000 epochs. Although the bertSEG model secured the highest COS score (0.830) and a noteworthy MSE score (0.059), it obtained a RANK score of 0.198 with 500 epochs. Overall, the ArBERTv2 embedding exhibited strong performance across different models, with Electra delivering the best results based on the RANK metric.

[1]https://huggingface.co/sentence-transformers/all-MiniLM-L12-v2

[2]https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2

[3]https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v2

Model	RANK	COS	MSE	epochs
Electra	0.035	0.823	0.098	1000
bertMSA	0.0009	0.937	0.079	1000
bertSEG	0.001	0.951	0.01	1000

Table 3: Results of the ArBERTv2 embedding on the RD task on the dev dataset. but the training was including the dev set.

Model	RANK	COS	MSE	epochs
Electra	**0.0644**	0.7071	0.158	500
bertMSA	0.1484	0.807	0.229	1000
bertSEG	0.198	0.830	0.059	500
$\text{BL}_C MSA$	1.09	0.818	0.219	-
$\text{BL}_C SEG$	1.26	**0.8436**	**0.055**	-
$\text{BL}_M elct$	0.84	0.736	0.145	-

Table 4: Results of the ArBERTv2 embedding on the RD task on the test dataset. Note that BL-C is the baseline results for CamelBERT model and MSA is bertMSA embedding and SEG is bertSEG. And the BL-M is the baseline of MARBERT model and elct is Electra embedding

4.2 WSD Results

Due to limited time, we conducted only a few experiments with the ArBERTv2 embedding for the WSD task, as shown in Table 4. Using the same model described in Section 3.3.2, we varied the number of epochs and the learning rate. Model 2 achieved the highest accuracy (0.268) and mrr@2 (0.297) with a learning rate of 1e-3 and 75 epochs. These preliminary results indicate promising directions for future research, warranting further investigation.

Model	Acc	mrr@2	lr	epochs
1	0.128	0.143	1e-4	100
2	**0.268**	**0.297**	1e-3	75
3	0.228	0.249	1e-3	50

Table 5: Results of the ArBERTv2 embeddings with WSD task on the test dataset

4.3 Error Analysis

In this section, we will analyze the outcomes and results by discussing two different aspects. First, we will examine the reasons why the ARBERT model outperformed other sentence transformer models. Second, we will evaluate the provided dataset, considering the common rule that "the quality of the data determines the quality of the model."

4.3.1 Models Analysis

The models used in this paper are the Arabic BERT model (ArBERTv2) and sentence transformers. BERT models, in general, have shown significant results in understanding language semantics. As noted by (Wang et al., 2024), "The bidirectional property of BERT-based models enables the learning of the context (i.e., semantic information) of unlabeled texts from both the left and the right sides. It lays the foundation for obtaining contextualized embeddings."

We believe this bidirectional nature is the main advantage of BERT models over sentence transformers, as their underlying targets and training approaches differ. According to (Reimers and Gurevych, 2019), the primary focus in training sentence transformers is to fine-tune BERT and RoBERTa models for sentence similarity tasks using cosine similarity with a pooling layer at the end. Additionally, the multilingual models provided by sentence transformers were trained in different languages, leading to a somewhat noisy embedding distribution due to language mixing. In contrast, the ArBERTv2 model was trained exclusively on a large and clean Modern Standard Arabic (MSA) text dataset, resulting in more expressive embeddings in our experiments.

Moreover, it is worth mentioning that the BERT-seg model (AraBERTv2) has shown better results in the MSE and cosine similarity criteria. These improved results can be attributed to the underlying ArBERTv2 model, which was used to extract the embeddings from the dataset.

4.3.2 Dataset Analysis

Research in the field of large language models (LLMs) has consistently shown that data quality is a major factor in the performance of these models (Naveed et al., 2024). To evaluate the quality of the given dataset, we conducted an analysis by randomly sampling 100 entries from each dataset segment. The main criticisms identified are as follows:

1. Lack of a systematic and logical methodology for definitions.

   The definitions vary significantly in their formulation. Some definitions focus on morphological forms, while others highlight a specific meaning derived from the general meaning of the Arabic root of the word. Additionally, some definitions are extracted from scientific contexts without specifying the targeted field, and some definitions use the word(s) intended to be defined within the definition itself, which is not a clear method of defining. examples are given in the next criticisms.

2. Reliance on word morphological forms rather than defining their meanings. Number of examples could be mentioned like تشمل where it only mentioned the where تشمُّلاً، فهو مُتشمِّل والمفعول مُتشمَّل به non of these are a meaning. Also some examples: ينوع, صعد.

3. Mentioning specific definitions without emphasizing the general absolute meaning. A lot of the definitions are written from a specific fields. For example: عرفي word they mentioned the legal definition. some other definitions the chosen meanings are very specific and narrowing the general meaning for example: بدن the definition بدن الثوب instead of the famous meaning 'body', الجسم. Also, وتر the definition is about the mathematical one وتر المثلث istead of the abstract famous meaning of the bow "وتر القوس حبله"

It is important for the dataset to be created in accordance with the logical rules of Arabic definitions, as described in both contemporary and classical texts on logic. Notable examples include "Tahzeeb Al Mantiq" (Refinement of Logic) by Al Taftazani and "Ta'rifaat" (Definitions) by Al Jurjani (Jurjānī, 1969). These works offer a systematic and logical framework for creating definitions, ensuring consistency and precision in the dataset, and it worth mentioning the field of the definition so the extraction get more detailed and specific For example in the "Ta'rifaat" book the word وجوب he mentioned four aspects of the definitions شرعي، أدائي، عقلي، فقهي.

5 Conclusion

In this paper, we conducted a comprehensive experimental analysis on sentence transformers and ArBERTv2 to create semantic embeddings for RD and WSD tasks, training a fixed neural network with four hidden layers. Our models achieved first place in both tasks, with a RANK score of 0.06 for the RD task and an accuracy score of 0.268 for the WSD task. Additionally, we performed a dataset analysis to develop a linguistic, logical methodology for building structured definitions for both tasks.

For future work, we plan to explore multiple methods for generating embeddings from the ArBERTv2 model and aim to build a larger and more comprehensive dataset to enhance our understanding of task semantics.

Acknowledgments

We would like to thank RIOTU labs at Prince Sultan University in Saudi Arabia for allowing us to use the lab server.

References

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Rawan Al-Matham, Waad Alshammari, Abdulrahman AlOsaimy, Sarah Alhumoud, Asma Wazrah, Afrah Altamimi, Halah Alharbi, and Abdullah Alaifi. 2023. KSAA-RD shared task: Arabic reverse dictionary. In *Proceedings of ArabicNLP 2023*, pages 450–460, Singapore (Hybrid). Association for Computational Linguistics.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. In *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, page 9.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2021. AraELECTRA: Pre-training text discriminators for Arabic language understanding. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 191–195, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2021. Consec: Word sense disambiguation as continuous sense comprehension. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1492–1503.

Nihed Bendahman, Julien Breton, Lina Nicolaieff, Mokhtar Boumedyen Billami, Christophe Bortolaso, and Youssef Miloudi. 2022. Bl. research at semeval-2022 task 1: Deep networks for reverse dictionary using embeddings and lstm autoencoders. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 94–100.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Ahmed Elbakry, Mohamed Gabr, Muhammad El-Nokrashy, and Badr AlKhamissi. 2023. Rosetta stone at KSAA-RD shared task: A hop from language modeling to word–definition alignment. In *Proceedings of ArabicNLP 2023*, pages 477–482, Singapore (Hybrid). Association for Computational Linguistics.

AbdelRahim Elmadany, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2022. Orca: A challenging benchmark for arabic language understanding. *arXiv preprint arXiv:2212.10758*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic bert sentence embedding. *Preprint*, arXiv:2007.01852.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, Kyiv, Ukraine (Online). Association for Computational Linguistics.

A.M. Jurjānī. 1969. *Kitab al-tarifat: a book of definitions*. Librairie du Liban.

Sanaa Kaddoura and Reem Nassar. 2024. Enhancedbert: A feature-rich ensemble model for arabic word sense disambiguation with statistical analysis and optimized data collection. *Journal of King Saud University-Computer and Information Sciences*, 36(1):101911.

Oscar Méndez, Hiram Calvo, and Marco A Moreno-Armendáriz. 2013. A reverse dictionary based on semantic analysis using wordnet. In *Advances in Artificial Intelligence and Its Applications: 12th Mexican International Conference on Artificial Intelligence, MICAI 2013, Mexico City, Mexico, November 24-30, 2013, Proceedings, Part I 12*, pages 275–285. Springer.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2024. A comprehensive overview of large language models. *Preprint*, arXiv:2307.06435.

Mohammad Taher Pilehvar. 2019. On the importance of distinguishing word meaning representations: A case study on reverse dictionary mapping. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2151–2156.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Rakia Saidi and Fethi Jarray. 2023. Sentence transformers and distilbert for arabic word sense induction. In *ICAART (3)*, pages 1020–1027.

Serry Sibaee, Samar Ahmad, Ibrahim Khurfan, Vian Sabeeh, Ahmed Bahaaulddin, Hanan Belhaj, and Abdullah Alharbi. 2023. Qamosy at Arabic reverse dictionary shared task: Semi decoder architecture for reverse dictionary with SBERT encoder. In *Proceedings of ArabicNLP 2023*, pages 467–471, Singapore (Hybrid). Association for Computational Linguistics.

Sushrut Thorat and Varad Choudhari. 2016. Implementing a reverse dictionary, based on word definitions, using a node-graph architecture. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2797–2806.

Jiajia Wang, Jimmy X. Huang, Xinhui Tu, Junmei Wang, Angela J. Huang, Md Tahmid Rahman Laskar, and Amran Bhuiyan. 2024. Utilizing bert for information retrieval: Survey, applications, resources, and challenges. *Preprint*, arXiv:2403.00784.