KSAA-RD Shared Task: Arabic Reverse Dictionary

Rawan Al-Matham[1*], Waad Alshammari[2*], Abdulrahman AlOsaimy[3], Sarah Alhumoud[4], Asma Al Wazrah[5], Afrah Altamimi[6], Halah Alharbi[7] and Abdullah Alfaifi[8]

King Salman Global Academy for Arabic Language (KSAA)

{[1]ralmatham, [2] walshammari, [3]aalosaimy, [4]salhumoud, [6]a.altamimi, [7]hmuneef, [8]aalfaifi}@ksaa.gov.sa

Abstract

This paper outlines the first KSAA-RD shared task, which aims to develop a Reverse Dictionary (RD) system for the Arabic language. RDs allow users to find words based on their meanings or definition. This shared task, KSAA-RD, includes two subtasks: Arabic RD and cross-lingual reverse dictionaries (CLRD). Given a definition (referred to as a "gloss") in either Arabic or English, the teams compete to find the most similar word embeddings of their corresponding word. The winning team achieved 24.20 and 12.70 for RD and CLRD, respectively in terms of rank metric. In this paper, we describe the methods employed by the participating teams and offer an outlook for KSAA-RD.

1 Introduction

A Reverse Dictionary (RD) is a type of dictionaries that allows users to find words based on their meanings or definitions. Unlike a traditional dictionary, where users search for a word by its spelling, a RD allow users to enter a description of a word or a phrase, and the RD will generate a list of words that match that description. RDs can be useful for writers, crossword puzzle enthusiasts, non-native language learners, and anyone looking to expand their vocabulary. Specifically, RD addresses the Tip-of-Tongue (TOT) phenomenon (Brown and McNeill, 1966), which refers to the situation where a person is aware of a word they want to say but is unable to express it accurately (Siddique and Sufyan Beg, 2019).

Various approaches have been proposed in the literature to develop RDs, including Information Retrieval (IR) System-based (Slaven et al., 2004; Crawford and Crawford, 1997; El-Kahlout and Oflazer, 2004; Shaw et al., 2013), Graph-based (Dutoit and Nugues, 2002; Reyes Magaña et al., 2019; Thorat and Choudhari, 2016), Mental Dictionary-based (Zock and Schwab, 2008; Zock and Bilac, 2004), Vector Space Model-based Semantic Analysis (Calvo et al., 2016; Méndez et al., 2013), and Neural Language Model-based approaches (Agrawal et al., 2021; Hedderich et al., 2019; Hill et al., 2016; Morinaga and Yamaguchi, 2018; Morinaga and Yamaguchi, 2020; Pilehvar, 2019; Qi et al., 2020; Yan et al., 2020; Zhang et al., 2020; Devlin et al., 2019).

However, to the best of our knowledge, there is no available Arabic RD system that allows the user to find the best matching word for a gloss in a specific dictionary, while most of the Arabic available digital dictionaries allow users to search for the definition by words (Siddique and Sufyan Beg, 2019).

We ran this shared task as a part of the first Arabic Natural Language Processing (ArabicNLP) conference collocated with EMNLP 2023, featuring the KSAA-RD (King Salman Global Academy for Arabic Language) with two subtasks: Arabic RD (Arabic to Arabic) and Cross-lingual Reverse Dictionary (CLRD) (Arabic to English). CLRD task aims to assist translating systems in selecting the best Arabic translation for new terms and definitions.

The dataset for both tasks used Arabic and English available dictionaries. Also, we provide manually annotated mapped dictionary between Arabic and English words to be used for supervised learning in the second task.

---

[*] Equal Contribution

A total of four papers submitted for the shared task. Three teams surpassed the RD task baseline, while all four teams exceeded the CLRD task baseline. We provide a description of all submitted systems and the approaches they use. All the datasets created for this shared task are publicly available to support further research in a GitHub repository[1].

The rest of this paper is organized as follows: Section 2 defines related work that tackled the RD problem. Section 3 presents shared task description and the subtasks included in KSAA-RD. Section 4 describes the data given in the task. Section 5 presents the methodology that is used to evaluate the performance of the systems. Section 6 provides the baseline system and its results, in addition to discussing the participating systems and their results in the shared task. Section 7 draws conclusions.

2 Related work

Various approaches have been proposed to develop RD systems, including Information Retrieval (IR) System-based Approach, Graph-based Approach, Mental Dictionary-based Approach, Vector Space Model-based Semantic Analysis Approach, and Neural Language Model-based Approach. The four subsections provide a preview for each approach respectively.

2.1 Information Retrieval (IR) System-based Approach

The traditional IR systems retrieve a ranked list of the most relevant words, and it has a long-standing tradition in computational semantics. An earliest work addressing reverse dictionary by (Crawford and Crawford, 1997) is a patented work that uses synonyms to enhance search capabilities, and provide a broader range of relevant words based on user queries.
Rather than searching the actual word, a study from (Slaven et al., 2004) analyzes the descriptions of target words and convert them into a structured representation. This structured representation allows for efficient matching and retrieval of words that closely match the given descriptions.
Another work from (El-Kahlout and Oflazer, 2004) explores a lexical database for retrieving

words based on their meanings. The method of extracting words based on their "meaning" involves comparing the user's definition with each entry in the Turkish database, without taking into account any semantic or grammatical information. The study from (Shaw et al., 2013) presents the development of a system that relies on a scalable database for efficient word retrieval. The system takes a user input phrase describing the desired concept and returns a word that satisfies the input phrase.

2.2 Graph-based Approach

A graph-based approach involves using a graph structure to represent the connections between words or concepts. This graph is built by considering semantic associations like synonyms, antonyms, hypernyms, and other related semantic links, to establish the relationships between the nodes representing the words or concepts. (Dutoit and Nugues, 2002) explores the connectivity and associations within the lexical database; by leveraging the graph structure to retrieve relevant words that align with the given definitions. Another approach focuses on utilizing the graph structure of a dictionary (Thorat and Choudhari, 2016) by investigating the sub-graph that surrounds each content word in a user query. They then prioritize and rank all the nodes encountered during the exploration, aiming to retrieve the most probable target word based on the given query. Another study from (Reyes Magaña et al., 2019) uses word association norms to establish semantic connections between words in the context of designing an electronic RD. The authors used the corpus of human-definitions and graph-based techniques, specifically a measure of betweenness centrality, to perform searches in the knowledge graph.

2.3 Mental Dictionary-based Approach

The mental dictionary-based approach depends on an individual's internal knowledge or mental lexicon to find words based on their meanings or descriptions. Instead of relying on external resources such as dictionaries or databases, this approach emphasizes using the individual's own mental representation of words and their

---

[1] https://github.com/Waadtss/ArReverseDictionary

associations. The study from (Zock and Bilac, 2004) proposes the concept of accessing words in an electronic dictionary by utilizing associations. This involves categorizing words based on the associations they evoke and identifying and labeling the most common or valuable associations within the dictionary. The proposal has been taken further in (Zock and Schwab, 2008) by implementing a user-guided search to the desired word that simulates human word synthesis, in order to gain a quick and intuitive access to that word.

2.4 Vector Space Model-based Semantic Analysis Approach

This approach attempts to use vector space models to transform the human-written queries into a vector by utilizing a semantic relations to improve the effectiveness of RD lookup. Another study that utilize semantic analysis with WordNet (Méndez et al., 2013) to generate vectors by identifying synsets that maximize a similarity measure. They then conduct a neighborhood search to extract the most relevant word. (Calvo et al., 2016) obtain vectors using LDA instead of WordNet.

2.5 Neural Language Model-based Approach

The neural language approach relies on encoding each input gloss into a vector representation, the output is a group of words whose embeddings are most similar to the corresponding gloss embedding. (Agrawal et al., 2021) enhance the traditional CBoW model by incorporating additional contextual information, such as word relationships and semantic associations, to better capture the nuances of word meanings. Another study that utilizes multi-sense embeddings (Hedderich et al., 2019) based on attention mechanism to enhance the representation of input queries in sentences. Focusing on eliminating the need for manually designed features, (Hill et al., 2016) propose Recurrent Neural Networks (RNN) to the RD task by encoding the definition of a word into a vector representation. The model then searches for the nearest neighbor word based on this vector. The performance was comparable to OneLook commercial RD. Another study from (Morinaga and Yamaguchi, 2018) improved the embedding accuracy by selecting better word vectors and employing category inference that

eliminate irrelevant results using Convolutional Neural Network (CNN). An approach from (Morinaga and Yamaguchi, 2020) aim to better capture the nuances of a word meaning and tackle the problem of sufficient capacities by combining the bidirectional long short-term memory (BiLSTM) with Cascade Forward Neural Network (CFNN). A neural model from (Zhang et al., 2020) which is a multi-channel RD model (MRDM) that consists of BiLSTM and attention as sentence encoder. The model can help find the target words by utilizing four characteristic predictors that predict the POS, morphemes, word category and sememes. (Pilehvar, 2019) incorporating more fine-grained representations by adopting sense embeddings to disambiguate senses of polysemous target words.

The BERT models were incorporated in RD tasks as well. (Devlin et al., 2019) employs BERT that capture the bidirectional contextual information of words and sentences, allowing it to better comprehend the context and meaning of language.

(Qi et al., 2020) develop an online RD that enhanced multi-channel RD model from (Zhang et al., 2020). The model uses BERT instead of BiLSTM as a sentence encoder. Another model from (Yan et al., 2020) use BERT in both monolingual and cross-lingual RD system.

To the best of our knowledge, this shared task is the first to target Arabic RD problem and there is no available Arabic RD system that allows the user to find the best word in a specific dictionary, while most of the Arabic available digital dictionaries allow the users to search for the definition by words (Siddique and Sufyan Beg, 2019).

Based on the previous studies and approaches, the neural language model-based approach gives promising results compared to other approaches due to its ability to map word embeddings for an input definition into an embedding of the word defined by the definition using neural networks. Such a function encodes phrasal semantics and bridges the gap between them and lexical semantics. Therefore, we applied it in our baseline.

3 Task Description

This section describes the two subtasks in detail: RD and CLRD. The former converts Arabic word definitions into Arabic embeddings, while the

latter converts English word definitions into Arabic embeddings.

3.1 Task 1: Reverse Dictionary

The structure of RDs (sequence-to-vector) is the opposite of traditional dictionaries lookup. This task focuses on the learning of how to convert human readable definitions into vector representation of the Arabic word.

id	ar.45
word	عين
POS	n
gloss	عضو الإبصار في الإنسان والحيوان.

(a) Example of definition in Arabic

```
{
  "id":"ar.45",
  "word":"عين",
  "gloss":"... عضو الإبصار في",
  "pos":"n",
  "electra":[0.4, 0.3, …],
  "sgns":[0.2, 0.5, …],
  "enId":"en.150",
  }
```

(b) Corresponding Arabic JSON snippet

id	en.150
word	eye
POS	n
gloss	One of the two organs in your face that are used for seeing

(c) Example of definition in English

```
{
  "id":"ar.45",
  "arword":"عين",
  "argloss":"... عضو الإبصار في",
  "arpos":"n",
  "electra":[0.4, 0.3, …],
  "sgns":[0.2, 0.5, …],
  "enId":"en.150",
  "word":"eye",
  "gloss":"One of the two ...",
  "pos":"n",
}
```

(d) Mapped JSON dictionary between Arabic and English languages

Figure 1: The structure of a data point.

In this task, the input for the model is an Arabic word definition (gloss) and the output is the corresponding Arabic word embedding. For instance, given the Arabic gloss "المسير ليلًا," the model would generate an embedding for the Arabic word "الإسراء" which is the word corresponding to the gloss.

The task involves reconstructing the word embedding vector of the defined word, rather than simply finding the target word that is similar to the approach used by (Mickus et al., 2022; Zanzotto et al., 2010; Hill et al., 2016). This would enable the users to search for words based on the definition or meanings they anticipate.

The training data collection contains a source word vector representation "electra and sgns" and its corresponding word definition "gloss", as illustrated in Figure 1 (a) and (b). The baseline model described in section 6.1 is designed to generate new word vector representations for the target unseen readable definitions.

3.2 Task 2: Cross-lingual Reverse Dictionary

The objective of the CLRDs task (sequence-to-vector) is to acquire the ability to transform readable definitions in the English language into a vector representation of the Arabic word. The main objective of this task is to identify the most accurate and suitable Arabic word vector that can efficiently express the identical semantic interpretation as the

provided English language definition or gloss, which is commonly known as Arabicization "تَعْريب".

In this task, the input for the model is an English word definition (gloss) and the output is the Arabic word embeddings corresponding to the gloss. For instance, given the English gloss "Travelling at night," the model would generate an embedding for the Arabic word "الإسراء."

The task involves reconstructing the word embedding vector that represents the Arabic word to its corresponding English definition. This approach enables users to search for words in other languages based on their anticipated meanings or definitions in English. This task facilitates cross-lingual search, language understanding, and language translation.

The data collection includes the word, source word vector representation "electra and sgns", and the definition "gloss" in both Arabic and English languages, as demonstrated in Figure 1 (d).

4   Data

This section discusses the data used in the shared task. The dataset includes two main components: the dictionary data, which is presented in section 4.1, and the word embedding vectors, which is presented in section 4.2. Section 4.3 describes further details of the dataset.

4.1   Dictionary data

To achieve the aim of the first task, known as RD, which seeks to develop a model capable of conducting reverse searches for Arabic words based on their meanings rather than their roots or lemmas, we utilized the Contemporary Arabic Language dictionary authored by Ahmed Mokhtar Omar (Omar, 2008). More specifically, we utilized the transferred version of this lexicon that adheres to the ISO standard, the Lexical Markup Framework (LMF) (Aljasim et al., 2022). It is worth mentioning that the KSAA team conducted this work. The dictionary relies on lemmas rather than roots, as discussed in the referenced study. The dataset comprises 58,000 words, commonly referred to as the lemmas that can have glosses with non-relevant information (e.g., morphological, and syntactic properties).

In the second task, our approach involved using a supplementary English dictionary, namely the English dictionary version employed in the SemEval 2022 Shared task on RD. It has a total of 63,596 lemmas that can have a different number of glosses (polysemy), and vice versa, a gloss can belong to more than one word (synonymy) (Mickus et al., 2022). This enabled us to construct a model that could effectively forecast the appropriate Arabic lemma matching to a given English meaning.

Consequently, there are two distinct datasets available: the dataset containing the Arabic dictionary and the dataset including the English dictionary. Each dataset consists of six components, including word form, part of speech, gloss, word ID, Electra embedding (Clark et al., 2020), and word2vec embedding (Mikolov et al., 2013). Within the realm of linguistic analysis, the "word" component encompasses the words. Additionally, the "part of speech" serves to denote the grammatical category to which the lemma belongs, namely noun, verb, adjective, adverb, or particle. The "gloss" serves the purpose of conveying the semantic content or meaning of a word, with the intention of excluding any phonetic, morphological, or syntactic aspects. The subsequent sections in the paper will provide explanations for the "electra" embedding and word2vec embedding components.

In order to fulfil the goals of the second task, we integrated the datasets in Arabic and English. The manual annotation procedure entailed a meticulous examination of the English gloss alongside its equivalent Arabic gloss, with the aim of attaining a thorough alignment between the two glosses across several linguistic dimensions.

To provide an instance, the English Dictionary defines the verb "cloud" as "to make obscure". This concept can be annotated with Arabic lemmas such as 'أغمض' which signifies the act of concealing, or 'أخفى', which denotes the act of covering, among other examples. The establishment of a correspondence between Arabic and English languages can be accomplished by assigning the Arabic gloss ID "id" to their corresponding English glosses "enId". Note that a word can have other irrelevant glosses (or meanings), but they will not be assigned.

The dictionary annotation process employs a systematic approach to facilitate manual annotation. For each entry in the English dictionary, *deep-translator*[2] library is employed to provide word translation "*wt*". Leveraging AraVec word embeddings (Soliman et al., 2017), the top ten similar word candidates are identified for *wt*. If any of these candidates align with lemmas in the Arabic dictionary, their corresponding IDs are integrated along with POS and gloss. The annotators then select the best candidate ID based on the corresponding POS and gloss that match the English dictionary.

In the manual phase, annotators meticulously select English lemmas, cross-referencing them with candidate IDs. This involves instances of confirmed matches, where corresponding Arabic lemma are included. In other cases, the process entails identifying the most suitable Arabic lemma translations within the Arabic dictionary and incorporating them. This meticulous process ensures data coherence, with lemma encompassing the word, POS, and gloss.

4.2 Embedding data

Our objective is to employ two distinct word embedding techniques, specifically contextualized word embedding and fixed word embedding. To efficiently attain this objective, we employ AraELECTRA (Antoun et al., 2021) for contextualized word embedding (referred to as "Electra"). AraELECTRA is an Arabic language representation model built upon the ELECTRA model. Unlike training a model to restore masked tokens, AraELECTRA focuses on training a discriminator model to distinguish original input tokens from replaced tokens, which have been substituted by a generator network. For single entries "*word*", we use the token's embedding; for multi-token entries, we average the token embeddings. This approach leverages the substantial volumes of high-quality language models that have already been trained, enabling us to harness the contextualized representation of existing large pretrained models. For fixed word embedding (referred to as skip-gram with negative sampling "sgns"), we employ the AraVec skip-gram architecture from (Soliman et al., 2017).

During the word2vec embedding extraction, a two-step approach is applied.

1. For a single-token *word*, a Unigrams skip-gram model is used to generate the embedding. When a *word* lacks representation (out-of-vocabulary) in the skip-gram model, the average embedding is obtained from the *gloss* associated with that word.

2. For a multi-token *word*, an N-Grams skip-gram model is employed to generate the embedding. When a multi-token *word* lacks representation (out-of-vocabulary) in the skip-gram model, the average embedding is obtained from the *gloss* associated with that word.

When a gloss lacks representation in the skip-gram model, the embedding is obtained from the stemmed gloss. The stem of each word in gloss is extracted using CAMEL tool (Obeid et al., 2020). When representation remains elusive, the Farasa stemmer (Darwish and Mubarak, 2016) is employed instead of CAMEL. The two-step approach is then employed for generating embeddings. When there is no representation, the stemmer is then employed at the word level.

4.3 Dataset description

The datasets are provided in JSON format comprising nearly 58k Arabic entry data points and 63k English data entry points. The dataset is divided into three splits, including a training split that consists of almost 78% of the data points, a validation split that consists of 11% of the data points, and a test split that consists of 11% of the data points. Refer to Table 1 for data statistics.

Task	Train	Dev	Test
RD	45,200	6,400	6,410
CLRD	2,843	299	1,213

Table 1: Data Statistics.

5 Evaluation

The primary objective of our tasks is to find the most similar Arabic embedding for an Arabic or English definition. Thus, we consider three different approaches to measure vector similarity. The first approach is a Mean Squared Error (MSE), which calculates the average squared difference between the generated reconstructed embedding and target embeddings. The second approach is using the cosine similarity measure, where a perfect reconstructed embedding would result in a cosine similarity of 1 with the target embeddings. The challenge with the cosine measure is that language models utilizing the Transformer architecture can produce anisotropic output. Hence, it is not reasonable to use it alone to anticipate that two random contextualized embeddings will be orthogonal (Ethayarajh, 2019; Timkey and van Schijndel, 2021).

To complement the limitations of both MSE and cosine measure, a third approach known as the ranking metric has been utilized. The ranking evaluation metrics proposed in the CODWOE SemEval competition (Mickus et al., 2022). As shown in equation (1), the ranking metric is concerned with comparing and evaluating the proportion between the reconstructed embedding cosine p_i and the target embedding cosine t_i to the reconstruction embedding cosine p_i with all other targets embedding t_j in the test set. The proportion of targets with a higher correlation is determined by identifying the number of cosine values greater than $\cos(p_i, t_i)$ (Mickus et al., 2022). The ranking metric can be described as:

$$\text{Ranking}(p_i) = \frac{\sum_{t_j \, Test \, set} \mathbb{1}_{\cos(p_i, t_j) > \cos(p_i, t_i)}}{\# \, Test \, set} \quad (1)$$

To select the top-performing and well-rounded model, the submitted systems evaluation process follows a hierarchy of metrics. The primary metric is the ranking metric, which is used to assess how well the model ranks predictions compared to ground truth values. If models have similar rankings, the secondary metric, mean squared error (MSE), is considered. Lastly, if further differentiation is needed, the tertiary metric, cosine similarity, provides additional insights.

6 Shared Task Teams & Results

In this section, we present our baseline model and, participating teams, and results and description of submitted systems.

6.1 Our Baseline system

The baseline architecture proposed by (Mickus et al., 2022) is based on the Transformer model introduced by (Vaswani et al., 2017). The architecture involves feeding the input **_gloss_**, which is represented as a sequence starting with a special token 'bos' and ending with another special token 'eos', into a straightforward Transformer encoder. The encoder generates hidden representations, which are then summed to produce the prediction. Additionally, a small non-linear feed-forward module is used to further refine the prediction. The evaluation of both tasks will be based on three different metrics including MSE, cosine similarity measure, and ranking metric.

6.2 Participating Teams

A total of 31 unique team registrations were received. A total of 39 valid submissions from 5 unique teams were received. During the testing phase, we received 5 submissions for the RD Subtask and 3 submissions for the CLRD Subtask from 4 different teams. You can find the details of these 4 teams in Table 2. Additionally, a total of 4 description papers were submitted and accepted.

Table 2: List of teams that participated

Team	Affiliation	Task
Rosetta Stone (ElBakry et al., 2023)	EPFL, Microsoft	1,2
UWB (Taylor, 2023)	University of West Bohemia	1,2
Qamosy (Sibaee et al., 2023)	Prince Sultan University	1
Abed (Qaddoumi, 2023)	NYU	1,2

6.3 Results and Description of Submitted Systems

Three teams participate in both RD and CLRD tasks, Rosetta Stone team, UWB teams, and Abed team. In the other hand, Qamosy team only submit the RD task. Results for both tasks are presented in Table 3 and Table 4.

The top team for both RD and CLRD tasks is the Rosetta Stone team (ElBakry et al., 2023). They employ an ensemble of fine-tuned Arabic BERT-based models, including camelBERT-MSA, camelBERT-Mix (Inoue et al., 2021), MARBERTv2 (Abdul-Mageed et al., 2021), and AraBERTv2 (Antoun et al., 2020). For RD, averaging the output embeddings from camelBERT-MSA and MARBERTv2 yielded a ranking of 24.20 using ELECTRA embeddings. For the CLRD task, they translate English glosses into Arabic, using the same models as in RD, achieving a rank of 12.70 with ELECTRA embeddings.

Qamosy team (Sibaee et al., 2023) methodology for RD task involves two phases: transforming the gloss into multidimensional vector representations using SBERT encoding, followed by training these vectors using the Simi-Decoder model. Their system achieved 2nd place in the RD task with a score of 28.10 in the Rank metric, utilizing ELECTRA embeddings.

Abed team (Qaddoumi, 2023) employs a modified multilingual BERT model for both RD and CLRD tasks, using data augmentation techniques like synonym replacement, random word insertion, deletion, and swapping in English, and random word deletion and swapping in Arabic. They achieved the 2nd and 3rd place in the RD and CLRD tasks, respectively, with a scores of 28.50 and 28.10 in the Rank metric with ELECTRA.

UWB teams (Taylor, 2023) utilize a rule-based approach for RD and CLRD tasks. They build a dataset-based dictionary and expand it using gloss. The dictionary-based approach with SGNS embeddings achieves 43.8 within RD task, lower than the baseline model, and 48.87 within the CLRD task.

It's evident that three teams utilizing BERT-based models outperformed our RD task Baseline, except for the dictionary-based approach by the UWB teams. However, the Rosetta Stone team's ensemble of different BERT architectures surpassed the performance of other methods.

Abed team demonstrated exceptional performance with ELECTRA embeddings, underscoring the substantial impact of data manipulation techniques on results. Surprisingly, the UWB teams, despite using a dictionary-based technique, outperformed our baseline that based on transformer model in the CLRD task.

7 Conclusion

In this paper, we present the first Arabic RD shared task, KSAA-RD, encompassing two subtasks: Arabic RD and Cross-Lingual RD, CLRD. The KSAA-RD task received 31 unique team registrations, resulting in 39 valid submissions and 4 submitted description papers. The outcomes from various teams underscore the persistent challenges posed by both RD and

CLRD tasks, emphasizing the need for continued research in the field of Arabic RD tasks.

Our experience with KSAA-RD emphasize the significant impact of data manipulation techniques. Furthermore, employing an ensemble of diverse transformer architectures proved superior to other methods, highlighting the importance of model diversity in enhancing performance.

The Arabic dictionary used in this shared task is limited, compared to the newly released dictionary of the Arabic contemporary language: "Alriyadh Dictionary" (KSAA, 2023), which contains more than 120K terms compared to 58K and it is manually verified by groups of experts in the KSAA. Other aspects of future work include exploring advanced embedding techniques that might be more suitable to the semantic notion of the problem. Future work includes employing these techniques in a search engine and analyzing the user behavior of the search results. Further investigation is needed to examine whether dictionary definitions (which are usually written in a formal style) are a good representation of the users' inquiries.

Table 3: Participants' results for Reverse Dictionary Track (RD)

	Embedding	Dev			Test		
		Cos ↑	MSE ↓	Rank ↓	Cos ↑	MSE ↓	Rank ↓
Baseline 200 epoch	Sgns	35.61	5.03	38.52 (3)	40.58	4.49	36.28 (4)
	Electra	48.84	24.94	31.27 (3)	50.79	23.04	31.87 (4)
Rosetta Stone	Sgns	55.19	3.45	28.12 (1)	60.50	3.00	**25.40 (1)**
	Electra	63.65	16.14	21.44 (1)	64.50	15.20	**24.20 (1)**
Qamosy	Sgns	---	---	---	39.40	6.50	30.80 (3)
	Electra	18.90	54.80	50.00 (4)	51.90	23.60	28.10 (2)
Abed team	Sgns	49.45	3.48	31.45 (2)	53.80	3.10	29.10 (2)
	Electra	61.69	16.75	24.90 (2)	62.50	15.70	28.50 (3)
UWB	Sgns	---	---	---	37.50	5.17	43.80 (5)
	Electra	---	---	---	---	---	---

*The primary metric for the evaluation is the rank – the lower the rank, the better the model

Table 4: Participants' Results for Cross-lingual Reverse Dictionary Track (CLRD)

	Embedding	Dev			Test		
		Cos ↑	MSE ↓	Rank ↓	Cos ↑	MSE ↓	Rank ↓
Baseline 300 epoch	Sgns	26.22	4.92	50.16 (3)	25.21	4.85	49.95 (4)
	Electra	54.09	22.10	36.22 (3)	51.66	23.81	40.72 (3)
Rosetta Stone	Sgns	38.74	4.84	37.15 (1)	40.00	5.30	**32.00 (1)**
	Electra	62.38	18.00	20.38 (1)	65.90	17.00	**12.70 (1)**
Abed team	Sgns	27.72	5.07	45.77 (2)	27.00	5.00	45.20 (2)
	Electra	58.06	19.55	25.88 (2)	56.50	20.60	28.10 (2)
UWB	Sgns	---	---	---	21.70	4.63	48.87 (3)
	Electra	---	---	---	---	---	---

*The primary metric for the evaluation is the rank – the lower the rank, the better the model

References

Muhammad Abdul-Mageed, AbdelRahim A. Elmadany, and El Moatez Billah Nagoudi. 2021. ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic. *CoRR*, abs/2101.01785. arXiv: 2101.01785.

Aarchi Agrawal, Athulkumar R, K S Ashin Shanly, Kavita Vaishnaw, and Mayank Singh. 2021. Reverse Dictionary Using an Improved CBoW Model. In *Proceedings of the 3rd ACM India Joint International Conference on Data Science & Management of Data (8th ACM IKDD CODS & 26th COMAD)*, page 420, New York, NY, USA. Association for Computing Machinery.

Hawra Aljasim, Sultanah Alghurabi, Halah Alharbi, Abdulrahman Alosaimy, Afrah Altamimi, Asyah Almutawa, and Abdullah Alfifi. 2022. Toward an Automatic Semantic Order of Word Meanings Based on Lexically Tagged Corpus. *Manuscript submitted for publication*.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based Model for Arabic Language Understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

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.

Roger Brown and David McNeill. 1966. The "tip of the tongue" phenomenon. *Journal of Verbal Learning and Verbal Behavior*, 5(4):325–337.

Hiram Calvo, Oscar Méndez, and Marco A. Moreno-Armendáriz. 2016. Integrated concept blending with vector space models. *Computer Speech & Language*, 40:79–96.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *CoRR*, abs/2003.10555. arXiv: 2003.10555.

H. Vance Crawford and Joel Crawford. 1997. Reverse electronic dictionary using synonyms to expand search capabilities.

Kareem Darwish and Hamdy Mubarak. 2016. Farasa: A New Fast and Accurate Arabic Word Segmenter. In *International Conference on Language Resources and Evaluation*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dominique Dutoit and Pierre Nugues. 2002. A lexical database and an algorithm to find words from definitions | Proceedings of the 15th European Conference on Artificial Intelligence. In *ECAI'02: Proceedings of the 15th European Conference on Artificial Intelligence*.

Ahmed ElBakry, Mohamed Gabr, Muhammad ElNokrashy, and Badr AlKhamissi. 2023. Rosetta Stone at KSAA-RD Shared Task: A Hop From Language Modeling To Word--Definition Alignment. In

˙Ilknur El-Kahlout and Kemal Oflazer. 2004. Use of Wordnet for Retrieving Words from Their Meanings. In *The Global Wordnet confrence*, pages 118–123.

Kawin Ethayarajh. 2019. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Michael A. Hedderich, Andrew Yates, Dietrich Klakow, and Gerard de Melo. 2019. Using Multi-Sense Vector Embeddings for Reverse Dictionaries. In *Proceedings of the 13th International Conference on Computational Semantics - Long Papers*, pages 247–258, Gothenburg, Sweden. Association for Computational Linguistics.

Felix Hill, Kyunghyun Cho, Anna Korhonen, and Yoshua Bengio. 2016. Learning to Understand Phrases by Embedding the Dictionary. *Transactions of the Association for Computational Linguistics*, 4:17–30.

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, pages 92–104, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

king Salman global academy for Arabic language KSAA. 2023. Alriyadh Dictionary.

Oscar Méndez, Hiram Calvo, and Marco A. Moreno-Armendáriz. 2013. A Reverse Dictionary Based on Semantic Analysis Using WordNet. In Félix Castro, Alexander Gelbukh, and Miguel González, editors, *Advances in Artificial Intelligence and Its Applications*, pages 275–285, Berlin, Heidelberg. Springer.

Timothee Mickus, Kees Van Deemter, Mathieu Constant, and Denis Paperno. 2022. Semeval-2022 Task 1: CODWOE – Comparing Dictionaries and Word Embeddings. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1–14, Seattle, United States. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 [cs].

Yuya Morinaga and Kazunori Yamaguchi. 2018. Improvement of Reverse Dictionary by Tuning Word Vectors and Category Inference. In Robertas Damaševičius and Giedrė Vasiljevienė, editors, *Information and Software Technologies*, pages 533–545, Cham. Springer International Publishing.

Yuya Morinaga and Kazunori Yamaguchi. 2020. Improvement of Neural Reverse Dictionary by Using Cascade Forward Neural Network. *Journal of Information Processing*, 28:715–723.

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. CAMeL Tools: An Open Source Python Toolkit for Arabic Natural Language Processing. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7022–7032, Marseille, France. European Language Resources Association.

Ahmed Mukhtar Omar. 2008. Dictionary of Contemporary Arabic Language. *Volume One, 1st Edition, Cairo, Egypt*:2109.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Abed Qaddoumi. 2023. Abed at KSAA-RD Shared Task: Enhancing Arabic Word Embedding with Modified BERT Multilingual. In

Fanchao Qi, Lei Zhang, Yanhui Yang, Zhiyuan Liu, and Maosong Sun. 2020. WantWords: An Open-source Online Reverse Dictionary System. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–181, Online. Association for Computational Linguistics.

Jorge C. Reyes Magaña, Gemma Bel Enguix, Gerardo Sierra, and Helena Gómez-Adorno. 2019. Designing an Electronic Reverse Dictionary Based on Two Word Association Norms of English Language. In *Electronic lexicography in the 21st century. Proceedings of the eLex 2019 conference. 1-3 October 2019, Sintra, Portugal, 2019, págs. 865-880*, pages 865–880. Lexical Computing.

Ryan Shaw, Anindya Datta, Debra VanderMeer, and Kaushik Dutta. 2013. Building a Scalable Database-Driven Reverse Dictionary. *IEEE Transactions on Knowledge and Data Engineering*, 25(3):528–540.

Serry Sibaee, Samar Ahmad, Ibrahim Khurfan, Vian Sabeeh, Ahmed Bahaaulddin, Hanan M. Belhaj, and Abdullah I. Alharbi. 2023. Qamosy at KSAA-RD shared task: Semi Decoder Architecture for Reverse Dictionary with SBERT Encoder. In

Bushra Siddique and Mirza Mohd Sufyan Beg. 2019. A Review of Reverse Dictionary: Finding Words from Concept Description. In Manish Prateek, Durgansh Sharma, Rajeev Tiwari, Rashmi Sharma, Kamal Kumar, and Neeraj Kumar, editors, *Next Generation Computing Technologies on Computational Intelligence*, pages 128–139, Singapore. Springer.

Bilac Slaven, Wataru Watanabe, Taiichi Hashimoto, Takenobu Tokunaga, and Hozumi Tanaka. 2004. Dictionary search based on the target word description. In pages 556–559.

Abu Bakr Soliman, Kareem Eissa, and Samhaa R. El-Beltagy. 2017. AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP. *Procedia Computer Science*, 117:256–265.

Stephen Taylor. 2023. UWB at KSAA-RD shared task: Computing the meaning of a gloss. In

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, Osaka, Japan. The COLING 2016 Organizing Committee.

William Timkey and Marten van Schijndel. 2021. All Bark and No Bite: Rogue Dimensions in Transformer Language Models Obscure Representational Quality. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4527–4546, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. *ArXiv*, abs/1706.03762.

Hang Yan, Xiaonan Li, Xipeng Qiu, and Bocao Deng. 2020. BERT for Monolingual and Cross-Lingual Reverse Dictionary. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4329–4338, Online. Association for Computational Linguistics.

Fabio Massimo Zanzotto, Ioannis Korkontzelos, Francesca Fallucchi, and Suresh Manandhar. 2010. Estimating Linear Models for Compositional Distributional Semantics. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1263–1271, Beijing, China. Coling 2010 Organizing Committee.

Lei Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2020. Multi-Channel Reverse Dictionary Model. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):312–319.

Michael Zock and Slaven Bilac. 2004. Word Lookup on the Basis of Associations : from an Idea to a Roadmap. In *Proceedings of the Workshop on Enhancing and Using Electronic Dictionaries*, pages 29–35, Geneva, Switzerland. COLING.

Michael Zock and Didier Schwab. 2008. Lexical access based on underspecified input. In *Coling 2008: Proceedings of the Workshop on Cognitive Aspects of the Lexicon (COGALEX 2008)*, pages 9–17, Manchester, United Kingdom. Coling 2008 Organizing Committee.