Evaluating RAG Pipelines for Arabic Lexical Information Retrieval: A Comparative Study of Embedding and Generation Models

Raghad Al-Rasheed, Abdullah Al Muaddi, Hawra Aljasim, Rawan Al-Matham, Muneera Alhoshan, Asma Al Wazrah, Abdulrahman AlOsaimy

{Ralrasheed@ksaa.gov.sa, Abdullah.AlMuadddi@gmail.com, haljasim@ksaa.gov.sa, ralmatham@ksaa.gov.sa, malhoshan@ksaa.gov.sa, aalwazrah@ksaa.gov.sa, aalosaimy@ksaa.gov.sa}

Abstract

This paper investigates the effectiveness of retrieval-augmented generation (RAG) pipelines, focusing on the Arabic lexical information retrieval. Specifically, it analyzes how embedding models affect the recall of Arabic lexical information and evaluates the ability of large language models (LLMs) to produce accurate and contextually relevant answers within the RAG pipelines. We examine a dataset of over 88,000 words from the Riyadh dictionary and evaluate the models using metrics such as Top-K Recall, Mean Reciprocal Rank (MRR), F1 Score, Cosine Similarity, and Accuracy. The research assesses the capabilities of several embedding models, including E5-large, BGE, AraBERT, CAMeLBERT, and AraELECTRA, highlighting a disparity in performance between sentence embeddings and word embeddings. Sentence embedding with E5 achieved the best results, with a Top-5 Recall of 0.88, and an MRR of 0.48. For the generation models, we evaluated GPT-4, GPT-3.5, SILMA-9B, Gemini-1.5, Aya-8B, and AceGPT-13B based on their ability to generate accurate and contextually appropriate responses. GPT-4 demonstrated the best performance, achieving an F1 score of 0.90, an accuracy of 0.82, and a cosine similarity of 0.87. Our results emphasize the strengths and limitations of both embedding and generation models in Arabic tasks.

1 Introduction

The rise in significance of machine learning and natural language processing (NLP) for tackling challenging linguistic tasks has led to notable progress in embedding and generation models (El-Beltagy and Abdallah, 2024; Chirkova et al., 2024). In English, many studies have explored the effectiveness of RAG and embedding models, demonstrating improvements in tasks like question-answering and information retrieval (Chirkova et al., 2024; Setty et al., 2024). However, in Arabic, fewer studies have addressed the unique challenges posed by its complex morphology and diacritics, which significantly affect model performance (Khondaker et al., 2024; Hijazi et al., 2024).

The primary objectives of this study are to evaluate the performance of various semantic embedding models for Arabic text retrieval and to assess the capabilities of large language models (LLMs) in performing question-answering tasks in Arabic using a retrieval-augmented generation (RAG) pipeline.

To achieve these goals, we conducted several experiments to address two key research questions: 1)How do different embedding models affect the recall of Arabic lexical information retrieval in RAG pipeline? 2)What is the best LLM for generating accurate and contextually relevant answers to Arabic lexical information questions within a RAG framework?

Our study goes further by focusing on extracting pertinent information from the Riyadh dictionary database, which includes more than 88,000 Arabic words . Embedding models are evaluated using metrics such as Recall@K and Mean Reciprocal Rank (MRR), while generation models are evaluated by accuracy, F1-score, and cosine similarity in answering context-specific questions. The study compares both closed-source and open-source models, including E5-large, AraBERT, CAMeLBERT, and AraELECTRA for embedding tasks, and GPT-4, GPT-3.5, SILMA-9B, Gemini-1.5, Aya-8B, and AceGPT-13B for generation tasks.

Our research provides valuable insights into the effectiveness of sentence embeddings versus word embeddings and explores how generation models manage semantic precision. Our findings aim to enhance the efficiency of NLP systems for Arabic.

The remainder of this paper is organized as follows: Section 2 presents the Literature Review, followed by the Methodology in Section 3. Sec-

Processings of the 1st Workshop on NLP for Languages Using Arabic Script (AbjadNLP), pages 155–164
January 19, 2025. ©2025 Association for Computational Linguistics

tion 4 provides a detailed description of the Dataset, while Section 5 discusses the Evaluation Dataset. The Results and Discussion are presented in Section 6, and finally, the study concludes with the Conclusion in Section 7.

2 Literature Review

Many studies have explored the effectiveness of retrieval-augmented generation (RAG) in enhancing large language models (LLMs) for tasks such as question-answering and information retrieval. by combining retrieval and generation techniques, these models produce more accurate and context-aware responses (Chirkova et al., 2024; Setty et al., 2024). Although these studies often focus on multilingual settings, they primarily concentrate on languages like English.

Research has highlighted the importance of embedding model selection for RAG systems, demonstrating that model similarity significantly impacts retrieval accuracy (Caspari et al., 2024; Montahaei et al., 2019). Additionally, semantic search plays a critical role in enhancing the relevance of generated content across various domains (Mahboub et al., 2024).

In the context of Arabic, research faces unique challenges due to the language's complex morphology and diverse dialects. Arabic-specific studies have begun to address these issues, particularly in the application of RAG. Benchmarks like LAraBench (Abdelali et al., 2024) and ArabLegalEval (Hijazi et al., 2024) demonstrate that dedicated Arabic models outperform general-purpose LLMs in tasks such as legal reasoning and sentiment analysis. However, the challenges posed by diacritics and dialect variation further complicate the optimization of RAG models (Khondaker et al., 2024). Diacritics, which are crucial for conveying meaning in written Arabic, have been largely overlooked in previous studies, leaving a gap in understanding their impact on model performance.

This study builds on prior research by evaluating a diverse set of open-source and proprietary models, including GPT-4, SILMA-9B, and E5-large, in the context of Arabic retrieval-augmented generation (RAG) pipelines. Using metrics such as Top-K Recall, MRR, F1 score, and cosine similarity, it provides a comprehensive performance comparison for Arabic lexical information retrieval and generation tasks. Additionally, the study examines over 88,000 Arabic words from the Riyadh dictionary, offering valuable insights into model capabilities for answering Arabic lexical information questions.

3 Methodology

The methodology involves a systematic, multi-step process as illustrated in Figure 1. The following sections provide detailed descriptions of the semantic embedding models, the vector indexing techniques, and the LLMs employed as generative models in this study.

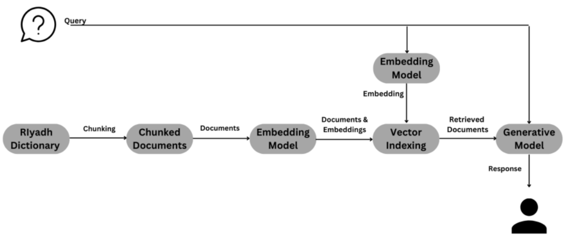


Figure 1: Illustration of the RAG methodology used in this study.

3.1 Corpus Preparation and Chunking

The initial step involves the preparation of the text corpus, with a focus on preserving the semantic integrity of the content. The corpus is segmented on a paragraph-by-paragraph basis, as described in Section 4, ensuring that the meaning and context within each paragraph are maintained. Each paragraph is restricted to a maximum of 512 tokens to comply with the token limit of the embedding model. In instances where a paragraph exceeds this limit, it is further divided into overlapping segments with an overlap of 50 tokens. This overlap maintains contextual continuity and ensures no critical information is lost during segmentation.

3.2 Embedding Models

Two types of embeddings are integrated in this study: word token embeddings with mean pooling, and sentence embeddings.These models were selected based on findings in previous research highlighted in the Section 2,

3.2.1 Word Embeddings

This approach involves generating embeddings for individual word tokens within a text, followed by applying a mean pooling layer to produce a single vector representation for each chunk of text. The models selected for this task include AraBERT v2 by (Antoun et al., 2020a), CAMeLBERT by Inoue et al. (2021), and AraELECTRA by Antoun et al. (2020b), chosen for their demonstrated effectiveness in handling Arabic text due to extensive pretraining on large-scale Arabic corpora.

- **AraBERT v2:** A transformer-based language model specifically designed for the Arabic language, AraBERT v2 has been trained on a vast corpus of Arabic text. Its architecture is based on the BERT model, adapted and fine-tuned to better address the linguistic characteristics of Arabic.

- **CAMeLBERT:** Part of the CAMeL toolkit, this model provides a comprehensive suite of Arabic NLP resources. CAMeLBERT is trained on a diverse set of Arabic dialects and formal texts.

- **AraELECTRA:** Using the ELECTRA pretraining approach, AraELECTRA focuses on learning through a discriminative model that identifies and corrects corrupted tokens in a text.

3.2.2 Sentence Embeddings

This approach involves generating embeddings for entire sentences or paragraphs, producing a single vector representation that captures the overall semantic content of the text. For this purpose, several models are selected:

- **E5-large:** A multilingual sentence embedding model developed by (Wang et al., 2022), E5-large is designed to generate high-quality semantic representations across multiple languages, including Arabic. It utilizes a text-to-text framework and is trained on a diverse range of tasks, including natural language inference, question answering, and semantic similarity.

- **Arabic-NLI-Matryoshka:** This model is a sentence-transformer finetuned from the AraBERT v2 base model on the Arabic NLI triplet dataset. It maps Arabic sentences and paragraphs to dense vectors, designed for tasks such as semantic textual similarity, semantic search, and text classification.

- **BGE (Big General Embeddings):** Originally developed to produce high-quality sentence embeddings for Chinese by (Xiao et al., 2023), the BGE model has also been trained on Arabic documents, thereby extending its applicability to Arabic text.

3.3 Vector Indexing

For the storage and retrieval of embedding vectors, this study employs FAISS (Facebook AI Similarity Search) by (Johnson et al., 2019), a well-known and efficient library designed for high-dimensional vector search. In this study, FAISS is utilized with the IndexFlatIP index, which leverages inner product calculations and the L2 distance metric to optimize the retrieval process. Additionally, cosine similarity is employed as the primary measure of similarity between vectors due to its effectiveness in capturing semantic relationships in high-dimensional spaces.

3.4 Generation

The final component of the methodology involves using LLMs as generative models for providing relevant answers to Arabic lexical information questions. After retrieving the most relevant documents from the vector store, a simple and clear prompt is used to provide context to the LLMs, as shown in Figure 2.To ensure the model follows the prompt exactly and generates deterministic outputs, the temperature parameter was set to 0 during all evaluations.

Original (Arabic)
إذا تم سؤالك عن كلمة معينة قم باستخراج الاجابة كما هي من النص من غير تغيير {context} {question}
Translation
If asked about a specific word, extract the answer exactly as it appears in the text without any changes. {context} {question}

Figure 2: Illustration of the prompt used

3.5 Corpus Preparation and Chunking

The study evaluates the performance of several LLMs, including GPT-3.5 (Ouyang et al., 2022), GPT 4o (OpenAI, 2023), Gemini-Flash-1.5 (Reid et al., 2024), AceGPT (Huang et al., 2023), Aya 8B (Aryabumi et al., 2024), and SILMA-9B-Instruct

(AI, 2023). These models were selected for their diversity in architecture, size, and pre-training, as well as their high ranking on the Arabic NLP leaderboard.[1] Furthermore, their inclusion was informed by findings from previous literature, which highlight their effectiveness in various Arabic natural language processing tasks such as text generation, sentiment analysis, and semantic understanding.

By evaluating this diverse set of LLMs, the study aims to provide insights into the most effective approaches for Arabic language generation within a retrieval-augmented pipeline to answer Arabic lexical information questions. The inclusion of models with high leaderboard rankings and evidence from prior research ensures that the study leverages state-of-the-art advancements in Arabic generative language models.

3.6 Embedding Models Evaluation

First, we evaluated the embedding models' ability to retrieve relevant context from 88,000 contexts within the Riyadh dictionary dataset, based on the provided question. The performance was assessed using **recall @K** (with k=1, k=3, and k=5) and **Mean Reciprocal Rank (MRR)**.

- **Recall @K Equation:**

$$Recall@K = \frac{\text{Number of relevant documents in top K}}{\text{Total number of relevent documents}} \tag{1}$$

- **Mean Reciprocal Rank (MRR) Equation:**

$$MRR = \frac{1}{|U|}\sum_{u=1}^{|U|}\frac{1}{rank_u} \tag{2}$$

These equations were used to measure how well the embedding models could identify the most relevant context for a given query.

3.7 Generation Models Evaluation

After retrieving the top 5 (k=5) potential contexts using the embedding models, the generation models were evaluated on their ability to select the correct context from these top candidates and generate accurate and contextually appropriate answers. This part of the evaluation tested how well the generation models could utilize the provided contexts to formulate coherent and correct answers.

The evaluations will utilize the following metrics:

- **F1 Score:** F1 Score: A perfect F1 score of 1 indicates optimal precision and recall, meaning all predictions were correct.

$$\text{F1 Score} = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{3}$$

- **Cosine Similarity:** A perfect cosine similarity score of 1 signifies that the reconstructed embedding is identical to the reference.

$$\text{Cosine Similarity} = \frac{\sum_{i=1}^{N} p_i q_i}{\sqrt{\sum_{i=1}^{N} p_i^2}\sqrt{\sum_{i=1}^{N} q_i^2}} \tag{4}$$

- **Accuracy:** This measures the percentage of correct predictions out of the total predictions made.

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{5}$$

Note: The dataset used for evaluation in 5 is unbalanced, which means that the performance of the generation models is assessed with special consideration to this characteristic. The evaluation metrics provide a comprehensive measure of the models' ability to select the correct context and generate accurate, coherent responses while accounting for challenges posed by an uneven distribution of data.

To ensure a fair evaluation of the models, the following micro-averaging formulas were used for F1-Score, Cosine Similarity, and Accuracy. Micro-averaging calculates the overall performance by considering the contributions of all instances equally, regardless of their class.

- **F1 Micro:** Computes the global F1 score by aggregating the contributions of all classes to precision and recall.

$$\text{F1}_{micro} = 2 * \frac{Precision_{micro} * Recall_{micro}}{Precision_{micro} + Recall_{micro}} \tag{6}$$

- **Cosine Similarity Micro:** Computes the overall cosine similarity by averaging across all instances.

$$CosineSimilarity_{micro} = \frac{\sum_{i=1}^{N} p_i q_i}{\sqrt{\sum_{i=1}^{N} p_i^2}\sqrt{\sum_{i=1}^{N} q_i^2}} \tag{7}$$

  where N is the total number of instances.

- **Accuracy Micro:** Computes the overall accuracy by considering all instances equally.

$$\text{Accuracy}_{micro} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{8}$$

[1] https://huggingface.co/spaces/OALL/Open-Arabic-LLM-Leaderboard

4 Dataset description

To compare the models used in the RAG pipeline, we used the Riyadh dictionary.[2]. The dataset comprises over 88,000 words, each including detailed information such as the stem, part of speech (POS), morphological pattern, non-diacritic lemma, definition (some words have multiple definitions, with a maximum of 31 definitions for a single word), translation, example, type, entry lemma of related words, and semantic field.

The data is structured and linked as shown in Figure 3 and Figure 4.

Original (Arabic)

الكلمة: [lemma]، وجذرها: [stem]، وهي [pos]
على وزن: [morphological Patterns]،
وشكلها بلا حركات: [nonDiacriticsLemma].

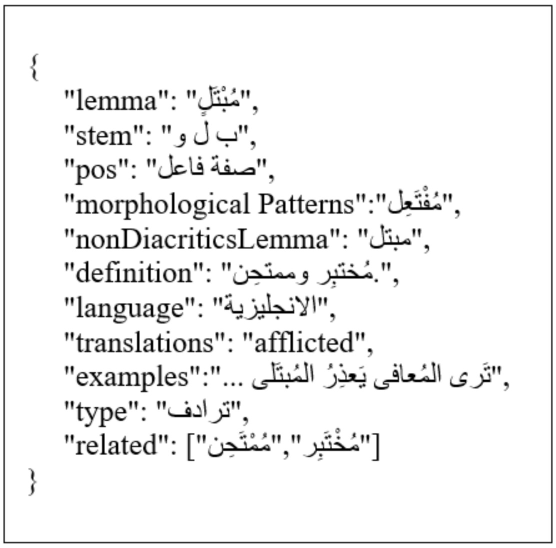
معنى الكلمة: [definition]،
ويقابلها في اللغة [translation]: [language].
ومن أمثلتها: [example]. للكلمة علاقة [type] بالكلمة: [related].

Translation

The word: [lemma], and its root: [stem], it is [pos]
in the pattern: [morphological Patterns],
and its form without diacritics is: [nonDiacriticsLemma].

The meaning of the word: [definition],
and its equivalent in [language]: [translation].
Examples include: [example].
The word has a [type] relation with the word: [related].

Figure 3: illustrate how The data is structured in the dataset.



```
{
    "lemma": "مُبْتَلٍ",
    "stem": "ب ل و",
    "pos": "صفة فاعل",
    "morphological Patterns":"مُفْتَعِل",
    "nonDiacriticsLemma": "مبتل",
    "definition": "مُختبِر وممتحِن.",
    "language": "الانجليزية",
    "translations": "afflicted",
    "examples":"تَرى المُعافى يَعذِرُ المُبتَلى ...",
    "type": "ترادف",
    "related": ["مُخْتَبِر","مُمْتَحِن"]
}
```

Figure 4: example that illustrates how the data was stored and linked.

5 Evaluation Dataset

The evaluation dataset includes 585 questions and answers distributed across eight categories, each targeting a specific linguistic aspect. These questions are based on 195 randomly selected words from the Riyadh dictionary and were meticulously crafted by Arabic linguists. The total number of questions per category is shown in Figure 5

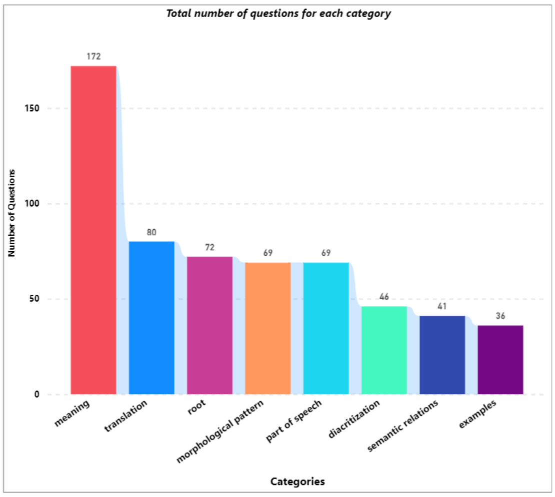


Figure 5: Total number of questions for each category in the evaluation dataset.

Each category targets a specific linguistic aspect:

- **Translation:** Involves translating words to and from Arabic.
- **Diacritization:** Focuses on accurately applying diacritical marks to ensure proper pronunciation and meaning of words.
- **Root:** Involves identifying the root forms of words.
- **Meaning:** Aims to provide definitions of words.
- **Morphological Pattern:** Examines the structural templates that define word forms.
- **Part of Speech:** Identifies the grammatical category of words.
- **Examples:** Identifying sentences that use words correctly.
- **Semantic Relations:** Explores relationships such as synonyms, antonyms, between words.

[2] https://dictionary.ksaa.gov.sa/

The evaluation of the RAG models involves two main components: assessing the embedding models ability to retrieve relevant context and evaluating the generation models performance in answering questions based on that context. The dataset includes ground truth context and answers developed by Arabic linguists, ensuring a reliable benchmark for these tasks.

6 Results and Discussion

In this section, we present the results of our study, focusing on the evaluation of two key areas: retrieval and generation LLMs.

The retrieval section examines the effectiveness of various embedding models in accurately identifying and retrieving relevant text segments from the Arabic dataset in Section 5. This part addresses the research question: How do different embedding models affect the recall of Arabic lexical information retrieval in RAG pipeline?

The generation section evaluates the performance of different LLMs in Arabic question-answering tasks, answers the question: What is the best LLM for generating accurate and contextually relevant answers to Arabic lexical information questions within a RAG pipeline?

6.1 Retrieval Embedding Models

The retrieval evaluation examined the capability of six semantic embedding models to accurately retrieve text segments that correspond to input queries. These models, representing both word embeddings with mean pooling and sentence embeddings, were tested on their ability to manage the complexities of Arabic text, particularly in the presence of diacritics. Performance was measured using Top-k Recall (k = 1, 3, 5) and MRR. The results, summarized in 1, reveal the performance differences among the evaluated models.

Model	**Top1**	**Top3**	**Top5**	**MRR**
E5	0.37	0.65	0.88	0.48
BGE	0.30	0.62	0.80	0.42
NLI	0.09	0.14	0.20	0.11
AraBERT v02	0.06	0.08	0.11	0.07
CamelBERT	0.04	0.10	0.16	0.06
AraElectra	0.02	0.06	0.09	0.04

Table 1: The table shows the Top-1, Top-3, and Top-5 Recall as well as MRR for each embedding model evaluated.

The E5 model demonstrated high performance across all metrics, achieving the highest scores in all Top-K Recall and MRR (0.48).This performance suggests that E5 effectively retrieves relevant context, with its architecture and training methodology being particularly well-suited for capturing the nuances of Arabic text.

BGE also showed strong performance, particularly in Top-3 (0.62) and Top-5 (0.80) Recall, indicating its capability to retrieve relevant information within a broader scope. However, its slightly lower Top-1 Recall (0.30) and MRR (0.42) compared to E5 suggest that while BGE is highly competitive, it may be less precise in consistently identifying the most relevant context.

A clear performance gap exists between E5, BGE, and the other models, particularly in Top-K Recall and MRR metrics. The reduced effectiveness of NLI, CamelBERT, AraBERT v02, and AraElectra in retrieving relevant segments suggests potential limitations in their model architectures or training data for this specific task.

The results indicate that sentence embeddings, particularly those produced by E5 and BGE, outperform word embeddings in the context of Arabic text. This suggests that sentence-level embeddings may be better suited for tasks requiring a comprehensive understanding of semantic content.

6.2 Generation with LLMs

To evaluate the performance of generation LLMs in answering Arabic lexical information questions, we evaluated various models using the dataset described in Section 5. The E5 model with k=5 context retrieval was selected to provide context based on our findings in Section 6.1.

The results in Table 2 summarizes the performance metrics of the evaluated LLMs. Presents a variations in performance across models and tasks. GPT-4o emerged as the top-performing model, achieving the highest overall micro F1-score 0.90 and micro accuracy 0.82, demonstrating its ability to generate accurate and relevant answers. SILMA-9B-Instruct excelled in micro cosine similarity 0.95, reflecting strong semantic alignment. Gemini-1.5 Flash performed robustly with a micro F1-score of 0.84 and micro accuracy of 0.72, while Aya 8B showed strength in micro cosine similarity 0.90 but exhibited lower micro F1-score 0.74 and micro accuracy 0.59, indicating its ability to capture semantic meaning but with reduced precision. GPT-3.5 displayed moderate performance,

Tasks	GPT-4o			Gemini-1.5-flash			SILMA-9B-Instruct			Aya 8B			GPT-3.5			AceGPT 13B		
	F1	Acc	Cos	F1	Acc	Cos	F1	Acc	Cos	F1	Acc	Cos	F1	Acc	Cos	F1	Acc	Cos
Translation	0.90	0.81	0.83	0.80	0.66	0.86	0.84	0.73	0.95	0.73	0.58	0.89	0.73	0.58	0.81	0.74	0.59	0.81
Diacritization	0.91	0.83	0.95	0.77	0.63	0.95	0.59	0.41	0.94	0.61	0.44	0.92	0.59	0.41	0.96	0.44	0.28	0.89
Root	0.99	0.97	0.85	0.96	0.93	0.85	0.97	0.94	0.99	0.96	0.93	0.95	0.97	0.94	0.86	0.96	0.92	0.85
Meaning	0.90	0.83	0.89	0.86	0.76	0.88	0.83	0.71	0.93	0.81	0.69	0.90	0.71	0.55	0.88	0.71	0.55	0.85
Morphological Pattern	0.98	0.96	0.86	0.94	0.88	0.86	0.91	0.84	0.99	0.80	0.67	0.94	0.87	0.77	0.86	0.82	0.70	0.85
Part of Speech	0.91	0.83	0.85	0.80	0.67	0.86	0.58	0.41	0.94	0.43	0.28	0.87	0.59	0.42	0.83	0.18	0.10	0.82
Examples	0.50	0.33	0.89	0.40	0.25	0.87	0.47	0.31	0.87	0.50	0.33	0.86	0.15	0.08	0.84	0.36	0.22	0.83
Semantic Relations	0.86	0.76	0.83	0.40	0.71	0.82	0.79	0.66	0.94	0.63	0.46	0.84	0.54	0.37	0.83	0.48	0.32	0.79
Average	**0.90**	**0.82**	**0.87**	**0.84**	**0.73**	**0.87**	**0.80**	**0.67**	**0.95**	**0.75**	**0.59**	**0.90**	**0.72**	**0.56**	**0.87**	**0.67**	**0.51**	**0.84**

Table 2: Model performance metrics across various tasks for different models. Metrics include F1-score (F1), Accuracy (Acc), and Cosine Similarity (Cos). The "Average" row represents the micro-average across all tasks.

with a micro F1-score of 0.72, micro accuracy of 0.56, and micro cosine similarity of 0.87, reflecting limitations in accuracy. AceGPT 13B was the weakest performer, with a micro F1-score of 0.67, micro accuracy of 0.51, and a relatively decent micro cosine similarity of 0.84. Despite being an Arabic-specific LLM, AceGPT 13B's precision and accuracy issues highlight significant gaps in its linguistic capabilities.

To evaluate the models' performance across eight distinct Arabic language processing tasks showed patterns in their capabilities and limitations within a RAG framework across different tasks. The analysis of semantic relations, diacritization, root extraction, meanings, morphological pattern recognition, part of speech tagging, example generation, and translation tasks shown in the Appendix A a sample of models responses across the tasks providing a thorough assessment of each model's linguistic capabilities.

Diacritization, which requires accurately applying Arabic vowel markers, proved challenging for most models. GPT-4o performed with the highest accuracy, closely aligning with the ground truth, achieving an F1-score of 0.91 and an accuracy of 0.83. For instance, in the task involving "أَسْهُمُ التَّأْسِيسِ", GPT-4o successfully applied the correct diacritics, producing "أَسْهُمُ التَّأْسِيسِ", distinguishing it from other models. In contrast, GPT-3.5 showed partial success, with an F1-score of 0.77 and accuracy of 0.63, but often applied diacritics inconsistently. For example, it produced partially diacritized outputs as"أَسْهُم التأسيس", failing to fully resolve ambiguities. Other models, including Gemini-1.5 Flash, SILMA-9B-Instruct, Aya 8B, and AceGPT 13B, frequently returned unmarked text, such as "اسهم التأسيس", as reflected in their lower F1-scores of 0.59–0.61 and accuracies of 0.28–0.44. This limitation stems from their training data and tokenizers, which do not prioritize diacritical information, resulting in outputs unsuitable for applications that depend on precise diacritic representation.

Conversely, root extraction appears as the highest-performing task, with all models achieving high F1-scores 0.957 to 0.986. The consistent accuracy across models demonstrated a steady handling of tasks requiring root extraction, exemplified as in the Appendix A by their correct identification of "ل و م" as the root of "ملوم".

The meanings task tested the models' ability to provide precise lexical definitions, where GPT-4o, Gemini-1.5 Flash, Aya 8B, and SILMA-9B-Instruct excelled by delivering definitions closely matching the ground truth. For instance, these models accurately defined “مُلَوِّم” as “مُوَبِّخٌ الشَّخْصَ مُعاتِبُهُ عَلى قَوْلٍ أَوْ عَمَلٍ غَيْرِ مُلائِمٍ” In contrast, GPT-3.5 and AceGPT 13B produced less accurate or overly verbose responses, underscoring their limitations in addressing tasks that demand lexical understanding.

Morphological pattern recognition, essential for understanding Arabic word structure, yielded accurate results across all evaluated models, with correct identification of the pattern "فُعَالَة" for "خُرَافَة". However, performance varied in consistency based on F1-score and accuracy.GPT-4o was the top performer, with an F1-score of 0.98 and accuracy of 0.96, consistently delivering precise outputs. Gemini-1.5 Flash and SILMA-9B-Instruct also performed strongly, achieving F1-scores of 0.94 and 0.91, with accuracies of 0.88 and 0.84. In comparison, AceGPT 13B and Aya 8B showed slightly lower performance, with F1-scores of 0.86 and 0.80 and accuracies of 0.82 and 0.67, respectively. These results emphasize the superior consistency of GPT-4o, Gemini-1.5 Flash, and SILMA-9B-Instruct, highlighting the impact of robust pretraining on morphological pattern recognition.

The translation task showed consistent performance across models, with most accurately translating terms like "مُشْتَبَه" to "suspect". Similarly, semantic relation identification, which assesses the ability to determine relationships between words or phrases, showed the strengths of SILMA-9B-Instruct and GPT-4o, as both models provided concise and accurate answers. For example, they correctly identified the relationship between "تَعْبِير" and "حُرِّيَّة اَلتَّعْبِيرِ" as collocation "تلازم". Gemini-1.5 Flash also demonstrated competence but occasionally included extraneous explanatory text. In contrast, GPT-3.5, Aya 8B, and AceGPT 13B struggled to accurately identify specific relationships, reflecting limitations in semantic reasoning.

POS tagging, a task requiring syntactic comprehension, revealed significant challenges for all models. Even GPT-4o, the leading performer, displayed inconsistencies in accuracy. Lower-performing models, such as GPT-3.5, Aya 8B, and AceGPT 13B, exhibited poor F1-scores and accuracy metrics. These results emphasize the need for refinement in Arabic-specific POS tagging. The most challenging task was generating accurate examples from the retrieved context, with GPT-4o achieving an F1-score of 0.50 and accuracy of 0.33 the highest among the models. Overall performance in this task, however, was suboptimal, with most models scoring below 0.50, underscoring the complexity of generative tasks in Arabic and the difficulty of synthesizing diverse, contextually appropriate examples.

This analysis of model performance across eight tasks highlights both strengths and limitations in the context of Arabic lexical information retrieval. GPT-4o consistently demonstrated superior performance, particularly in semantic reasoning and diacritization, while SILMA-9B-Instruct showed its ability to maintain semantic consistency . Gemini-1.5 Flash delivered reliable results across multiple tasks. On the other hand, models such as GPT-3.5, Aya 8B, and AceGPT 13B struggled with precision and linguistic understanding.

6.3 Adapting the RAG Pipeline for Abjad and Ajami Languages

The findings from this study on RAG for Arabic lexical retrieval can be extended to languages like Pashto, Sindhi, and Uyghur, as GPT-4 and Gemini-1.5 Flash already support these languages through multilingual. Their ability to handle morphologically complex languages such as Arabic and Persian suggests strong potential for processing similar languages that use Abjad or Ajami scripts.

The RAG pipeline discussed in this study could be adapted for these languages by leveraging its strengths in semantic representation and contextual generation. The embedding model E5, known for its multilingual support, already covers Persian and could be extended to Pashto, Sindhi, and Uyghur with additional pretraining on relevant datasets(Wang et al., 2022).

Adapting the RAG pipeline would require addressing specific challenges such as handling diacritics in Pashto and Sindhi, tone markings in Uyghur, and limited digital corpora for these languages. Transfer learning from Arabic and Persian models could mitigate these limitations, while customized tokenization methods tailored to Abjad and Ajami scripts could improve retrieval and generation tasks. Future research should explore expanding model capabilities through multilingual and script-specific fine-tuning.

7 Conclusion

This study evaluates the performance of embedding models in the recall of Arabic lexical information retrieval and LLMs in processing and generating relevant answers to Arabic lexical information questions. The results show that sentence embedding models like E5 outperform in retrieval tasks, achieving high accuracy in capturing semantic relationships. For generation tasks, models such as GPT-4o, Gemini-1.5 Flash, and SILMA-9B-Instruct perform strongly, with GPT-4o leading in generative capabilities. However, challenges remain in areas like diacritization and part-of-speech tagging, where models like GPT-3.5 and AceGPT 13B showed limitations. Future work should focus on optimizing these models and expanding datasets to improve their handling of complex Arabic linguistic features.

References

Ahmed Abdelali, Hamdy Mubarak, Shammur Absar Chowdhury, Maram Hasanain, and Ahmed Ali. 2024. Larabench: Benchmarking arabic ai with large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics*.

SILMA AI. 2023. Silma-9b-instruct-v1.0. `https://huggingface.co/silma-ai/SILMA-9B-Instruct-v1.0`.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020a. Arabert: Transformer-based model for arabic language understanding. *arXiv preprint arXiv:2003.00104*.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020b. Araelectra: Pre-training text discriminators for arabic language understanding. *arXiv preprint arXiv:2012.15516*.

Vashista Aryabumi, James Dang, Dhanusha Talupuru, Sarvesh Dash, Daniel Cairuz, Harrison Lin, et al. 2024. Aya 23: Open weight releases to further multilingual progress. *arXiv preprint arXiv:2405.15032*.

Laura Caspari, Kanishka Ghosh Dastidar, Saber Zerhoudi, Jelena Mitrovic, and Michael Granitzer. 2024. Beyond benchmarks: Evaluating embedding model similarity for retrieval augmented generation systems. *arXiv preprint arXiv:2407.08275*.

Nadezhda Chirkova, David Rau, Hervé Déjean, Thibault Formal, Stéphane Clinchant, and Vassilina Nikoulina. 2024. Retrieval-augmented generation in multilingual settings. *arXiv preprint arXiv:2407.01463*.

Samhaa R. El-Beltagy and Mohamed A. Abdallah. 2024. Exploring retrieval augmented generation in arabic. In *Procedia Computer Science*.

Faris Hijazi, Somayah AlHarbi, Abdulaziz AlHussein, et al. 2024. Arablegaleval: A multitask benchmark for assessing arabic legal knowledge in large language models. *arXiv preprint arXiv:2408.07983*.

Haoyang Huang, Feng Yu, Jiangzhe Zhu, Xiaowen Sun, Hao Cheng, Dawei Song, et al. 2023. Acegpt, localizing large language models in arabic. *arXiv preprint arXiv:2309.12053*.

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. *arXiv preprint arXiv:2103.06678*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Md Tawkat Islam Khondaker, Numaan Naeem, Fatimah Lyba Khan, AbdelRahim A. Elmadany, and Muhammad Abdul-Mageed. 2024. Benchmarking llama-3 on arabic language generation tasks. In *Proceedings of The Second Arabic Natural Language Processing Conference*.

Ali Mahboub, Muhy Eddin Za'ter, et al. 2024. Evaluation of semantic search and its role in retrieved-augmented-generation (rag) for arabic language. *arXiv preprint arXiv:2403.18350*.

Ehsan Montahaei, Danial Alihosseini, and Mahdieh Soleymani Baghshah. 2019. Jointly measuring diversity and quality in text generation models. *arXiv preprint arXiv:1904.03971*.

OpenAI. 2023. GPT-4 Technical Report. Technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *CoRR*.

Machel Reid, Nikolay Savinov, Dmitry Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-Baptiste Alayrac, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Spurthi Setty, Harsh Thakkar, Alyssa Lee, Eden Chung, and Natan Vidra. 2024. Improving retrieval for rag based question answering models on financial documents. *arXiv preprint arXiv:2404.07221*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Ranjan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding.

A Model Responses for different Task

Tasks	Question	Ground Truth Answer	SILMA-9B-Instruct	Aya 8B	Gemini 1.5 Flash	GPT4o	GPT3.5	AceGPT 13B
Semantic Relations	ما العلاقة الدلالية الرابطة بين (تَعْبِير) بـ (حُرِّيَّة اَلتَّعْبِيرِ)؟	تلازم	تلازم	العلاقة الدلالية بين (تَعْبِير) و (حُرِّيَّة اَلتَّعْبِيرِ) هي أن الأولى هي اسم والثاني هو اسم آخر يصف المعنى الأول.	العلاقة الدلالية الرابطة بين (تَعْبِير) بـ (حُرِّيَّة اَلتَّعْبِيرِ) هي التلازم.	العلاقة الدلالية الرابطة بين (تَعْبِير) و (حُرِّيَّة اَلتَّعْبِيرِ) هي علاقة تلازم.	العلاقة الدلالية الرابطة بين "تعبير" و "حرية التعبير" هي أن "تعبير" يشير إلى الفعل أو العملية التي تقوم بها الحرية التعبير.	العلاقة الدلالية الرابطة بين "تعبير" و "حرية التعبير" هي أن "تعبير" يشير إلى الفعل أو العملية التي تقوم بها الحرية التعبير.
Diacritization	اكتب (أسهم التأسيس) مع تشكيلها بالحركات	أَسْهُم اَلتَّأْسِيسِ	أسهم التأسيس	أسهم التأسيس	أَسْهُم اَلْتَّأْسِيسِ، بتشكيلها: أسهم التأسيس	أَسْهُمُ التَّأْسِيسِ	أَسْهُم اَلتَّأْسِيسِ	أسهم التأسيس
Root	ما الجذر اللغوي لكلمة (ملوم)؟	ل و م	ل و م	ل و م	الجذر اللغوي لكلمة (ملوم) هو: ل و م	الجذر اللغوي لكلمة (ملوم) هو: ل و م.	الجذر اللغوي لكلمة "ملوم" هو "ل و م."	الجذر اللغوي لكلمة (ملوم) هو (ل و م).
Meaning	ما معنى كلمة (مُلَوِّم)؟	مُوَبِّخٌ الشَّخْصَ مُعاتِبُهُ عَلى قَوْلٍ أَوْ عَمَلٍ غَيْرِ مُلائِمٍ.	هو رأي سلبي	الإجابة: مُوَبِّخٌ الشَّخْصَ مُعاتِبُهُ عَلى قَوْلٍ أَوْ عَمَلٍ غَيْرِ مُلائِمِ	معنى كلمة (مُلَوِّم) هو: مُوَبِّخٌ الشَّخْصَ مُعاتِبُهُ عَلى قَوْلٍ أَوْ عَمَلٍ غَيْرِ مُلائِمِ	معنى كلمة (مُلَوِّم) هو: مُوَبِّخٌ الشَّخْصَ مُعاتِبُهُ عَلى قَوْلٍ أَوْ عَمَلٍ غَيْرِ مُلائِمٍ.	مُلَوِّم: مُوَبِّخٌ الشَّخْصَ مُعاتِبُهُ عَلى قَوْلٍ أَوْ عَمَلٍ غَيْرِ مُلائِمٍ.	معنى كلمة (مُلَوِّم) هو مُوَبِّخ الشَّخْصَ مُعاتِبُهُ عَلى قَوْلٍ أَوْ عَمَلٍ غَيْرِ مُلائِمٍ. هذه الكلمة تعني الشخص الذي يُلام أو يُوبخ شخصًا آخر بسبب قول أو عمل غير مناسب.
Part of Speech	ما القسم الكلامي لكلمة (تَعَاتَبْ)؟	فعل	فعل أمر	الجواب: فعل أمر	فعل أمر	القسم الكلامي لكلمة (تَعَاتَبْ) هو: فعل.	القسم الكلامي لكلمة (تَعَاتَبْ) هو فعل.	القسم الكلامي لكلمة (تَعَاتَبْ) هو الفعل.
Examples	أعطني مثالاً على استخدام كلمة (أوبال).	أوبالٌ حَجَرٌ كريمٌ نَصْفُ شَفَّافٍ بِأَلوانٍ مُتَعَدِّدَةٍ	أوبالٌ حَجَرٌ كريمٌ نَصْفُ شَفَّافٍ بِأَلوانٍ مُتَعَدِّدَةٍ.	مثال عليها: أوبالٌ حَجَرٌ كريمٌ نَصْفُ شَفَّافٍ بِأَلوانٍ مُتَعَدِّدَةٍ.	أوبالٌ حَجَرٌ كريمٌ نَصْفُ شَفَّافٍ بِأَلوانٍ مُتَعَدِّدَةٍ.	أوبالٌ حَجَرٌ كريمٌ نَصْفُ شَفَّافٍ بِأَلوانٍ مُتَعَدِّدَةٍ.	عندما رأيت الأوبال الجميل في محل المجوهرات، قررت شراؤه لصديقتي كهدية.	أوبالٌ حَجَرٌ كريمٌ نَصْفُ شَفَّافٍ بِأَلوانٍ مُتَعَدِّدَةٍ.
Translation	ما ترجمة كلمة (مُشْتَبَه) بالإنجليزية؟	Suspect	Suspect	Suspect	Suspect	ترجمة كلمة (مُشْتَبَه) بالإنجليزية هي : Suspect.	Suspect	ترجمة كلمة (مُشْتَبَه) بالإنجليزية هي "suspect".
Morphological Pattern	ما الوزن الصرفي لكلمة (خُرَافَة)؟	فُعَالَة	فُعَالَة	الإجابة: فُعَالَة	الوزن الصرفي لكلمة (خُرَافَة) هو: فُعَالَة	الوزن الصرفي لكلمة (خُرَافَة) هو: فُعَالَة.	الوزن الصرفي لكلمة "خُرَافَة" هو فُعَالَة.	الوزن الصرفي لكلمة (خُرَافَة) هو (فُعَالَة).