Benchmarking Embedding Models for Persian-Language Semantic Information Retrieval

1st Mahmood Kalantari
Computer and Science Engineering
IUST
Tehran, Iran
m_kalantari76@comp.iust.ac.ir

2nd Mehdi Feghhi
Computer and Science Engineering
IUST
Tehran, Iran
feghhi_me@comp.iust.ac.ir

3rd Nasser Mozayani
Computer and Science Engineering
IUST
Tehran, Iran
mozayani@iust.ac.ir

***Abstract*—The increasing reliance on semantic-based retrieval, especially in the context of large language model-powered chatbots, underscores the need for robust evaluation of embedding models. In this study, the performance of embedding models for Persian-language information retrieval was investigated, addressing an area with limited prior research. Four question-answering datasets were used—two publicly available datasets adapted for this study and two custom datasets derived from translations. A systematic evaluation of 17 embedding models was conducted, and the models were ranked based on their accuracy in retrieving relevant content using similarity measures such as dot product, cosine similarity, and L2 distance. The findings emphasize the adaptability of these models to diverse textual data and address the specific challenges posed by the Persian language. This research bridges a critical gap in Persian-language retrieval tasks, providing a comprehensive benchmark for evaluating embedding models in semantic information retrieval scenarios.**

***Index Terms*—Embedding search, Embedding models, Persian embedding, Persian question-answering, Retrieval-Augmented Generation (RAG)**

I. Introduction

With the growing adoption of chatbots powered by large language models (LLMs) known for their advanced text-generation capabilities, notable challenges have surfaced, including the tendency to provide incomplete or inaccurate answers in specialized domains and the generation of misleading information, often referred to as ”hallucinations.” To address these issues, Retrieval-Augmented Generation (RAG) has been developed [1] [2], a method that enhances response accuracy by integrating relevant information from external sources. Additionally, the limited context size of LLMs restricts their ability to analyze extensive documents effectively, making precise conclusions difficult [3]. Overcoming this requires robust information retrieval, which relies on embedding models to create accurate vector representations that help identify semantically relevant content [4].

The RAG framework leverages information retrieval and semantic search techniques to extract the most relevant content. These techniques are underpinned by semantic embeddings, created using encoder models. By comparing the semantic similarity between a question and potential answers, the system identifies the most relevant information [5]. Since the quality of the responses heavily depends on the quality of these embeddings, it is crucial to evaluate and compare different embedding models systematically.

Our work emphasizes evaluating and comparing embedding models for Persian-language information retrieval, an area that has been largely underexplored. While previous studies have primarily applied embeddings to tasks such as text classification [6], sentiment analysis [7], and translation [8], limited attention has been given to their use in retrieving relevant context from large documents.

This study introduces a framework for evaluating embedding models using four diverse question-answering datasets, including two publicly available datasets adapted for specific needs and two custom datasets created by translating existing content. Seventeen embedding models were tested and ranked based on their ability to retrieve relevant content accurately. By employing datasets designed to retrieval scenarios, this comprehensive evaluation highlights the adaptability of the models across various types of textual data. Addressing a critical gap in Persian-language retrieval tasks, this study assesses embedding models' effectiveness in identifying relevant content while accounting for the unique complexities of the Persian language.

The paper is structured as follows: it begins with a detailed discussion of the methodology used to evaluate embedding models, including techniques for calculating similarity (dot product, cosine similarity, and L2 distance) and key evaluation metrics. This is followed by an overview of the datasets employed and concludes with a comparative analysis of the results.

II. Related Work

Recent developments in embedding models underscore their essential role in various natural language processing (NLP) tasks, especially when combined with advanced systems like large language models (LLMs) and retrieval-augmented generation (RAG). As these models are fine-tuned, several studies have delved into the key challenges surrounding context management, the integration of external knowledge, and task-specific optimization [10].

Chen et al. (2023) conducted a study to benchmark LLMs within RAG frameworks, with the aim of assessing their capacity to handle noisy data, incorporate external knowledge, and

979-8-3315-2225-4/24/$31.00 ©2024 IEEE

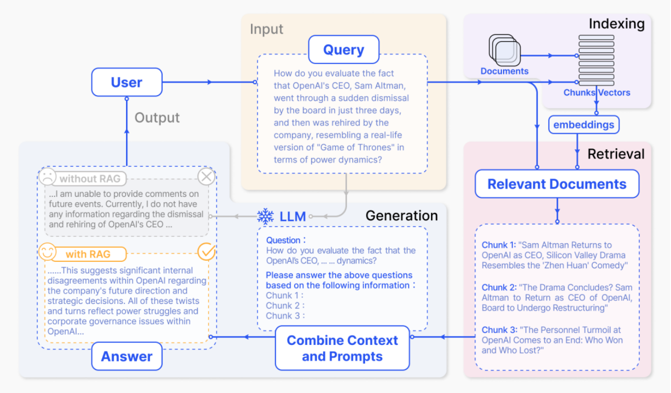

Input
Query
How do you evaluate the fact that OpenAI's CEO, Sam Altman, went through a sudden dismissal by the board in just three days, and then was rehired by the company, resembling a real-life version of "Game of Thrones" in terms of power dynamics?
Indexing
Documents
Chunks
Vectors
embeddings
User
Output
Retrieval
Relevant Documents
without RAG
...I am unable to provide comments on future events. Currently, I do not have any information regarding the dismissal and rehiring of OpenAI's CEO ...
LLM
Generation
with RAG
......This suggests significant internal disagreements within OpenAI regarding the company's future direction and strategic decisions. All of these twists and turns reflect power struggles and corporate governance issues within OpenAI...
Question :
How do you evaluate the fact that the OpenAI's CEO, ... dynamics?
Please answer the above questions based on the following information :
Chunk 1 :
Chunk 2 :
Chunk 3 :
Chunk 1: "Sam Altman Returns to OpenAI as CEO, Silicon Valley Drama Resembles the 'Zhen Huan' Comedy"
Chunk 2: "The Drama Concludes? Sam Altman to Return as CEO of OpenAI, Board to Undergo Restructuring"
Chunk 3: "The Personnel Turmoil at OpenAI Comes to an End: Who Won and Who Lost?"
Answer
Combine Context and Prompts

Fig. 1. A representative instance of the RAG process applied to question answering. It mainly consists of 3 steps. 1) Indexing. Documents are split into chunks, encoded into vectors, and stored in a vector database. 2) Retrieval. Retrieve the hit@k chunks most relevant to the question based on semantic similarity. 3) Generation. Input the original question and the retrieved chunks together into LLM to generate the final answer [9].

avoid inaccuracies. While RAG has been shown to improve certain aspects of model performance, the study also brought attention to notable shortcomings, such as the difficulty LLMs encounter when tasked with synthesizing information from multiple sources or identifying factual errors. These findings emphasize the need for more robust RAG systems that can effectively deal with incomplete or noisy inputs [11] .

In another study, Zhang et al. (2023) addressed the limitations of LLMs by introducing LLM-Embedder, a unified retrieval-augmented embedding model. This model was designed to streamline the retrieval process, enabling LLMs to access external information with greater flexibility and efficiency. By employing reward-based optimization and multi-task fine-tuning, LLM-Embedder significantly improved the model's ability to retrieve relevant data, manage memory, and utilize external tools. This research pointed to the importance of unified embedding models that can handle diverse tasks without depending on specialized retrievers for each [10].

To further explore RAG's constraints, a new study introduced MultiHop-RAG, along with a dataset designed to assess retrieval systems' ability to handle multi-hop reasoning tasks. These queries, which require pulling information from multiple sources and making connections across them, presented major difficulties for existing LLMs and retrieval systems. The MultiHop-RAG dataset, focusing on inference, comparison, and temporal queries, showed that while current embedding models perform well on single-hop queries, they still face challenges in synthesizing relevant information across multiple documents [12].

A recent survey mapped the evolution of RAG systems, tracing their development from simple "retrieve-read" frameworks to more sophisticated modular designs. This survey highlighted key innovations, such as the use of modular systems that integrate flexible memory and routing components to improve adaptability and accuracy across a wider range of tasks. This shift towards modularity is crucial for enabling RAG systems to handle complex real-world applications, including long-term memory management and responses to adversarial inputs [13].

In addition, research on embedding models for low-resource languages, such as Persian, continues to highlight the need for task-specific evaluation benchmarks. A study focusing on Persian word embeddings introduced new benchmarks for tasks like analogy and concept categorization, emphasizing the role of embedding models in capturing semantic relationships in underrepresented languages. This work provides an important foundation for the development of NLP tools for these linguistic communities [14].

Collectively, these studies demonstrate the broad applications and ongoing challenges faced by embedding models, especially in retrieval-augmented tasks. As these models continue to develop, future research will need to focus on integrating longer contexts, improving retrieval for multi-document queries, and extending high-quality embeddings to support low-resource languages.

III. Embeddings and Dataset

A. *Embeddings*

Embedding models have become a cornerstone in the semantic analysis of texts, with tremendous progress made over the years. Early breakthroughs in this area were achieved with the introduction of models like Word2Vec and GloVe [15] [16]. Word2Vec, through techniques like Skip-gram and CBOW, created vector representations that captured the relationships between words. GloVe, on the other hand, used a co-occurrence matrix to generate distributed semantic embeddings.

More advanced models soon followed, such as FastText, which improved on previous approaches by incorporating subword information [17]. This made it particularly effective for dealing with complex languages and unknown words. However, the development of embedding models for non-English languages, like Persian, lagged behind until more recently.

As the demand for natural language processing (NLP) models tailored to Persian grew, several embedding models specifically for the language were developed:

- Farsi Word2Vec: Built on the Word2Vec approach, this model generates semantic embeddings for Persian words by training on Persian-language datasets like newspapers and books [18].
- FastText Persian: FastText's version for Persian, designed to handle the language's intricate structure and word combinations more effectively.
- ParsBERT: A Persian-specific adaptation of the BERT [19] architecture, trained on a large corpus of Persian text. ParsBERT has performed well in tasks such as text classification and sentiment analysis. A more advanced variation of ParsBERT, such as the one from Hooshvare, has also been trained [20].

- MiniLM: A lighter version of models like BERT and RoBERTa [21], introduced by Microsoft. It reduces computational costs without sacrificing too much accuracy, making it suitable for tasks like machine translation and question answering [22].
- Sentence-BERT (SBERT): Built for comparing sentences semantically, SBERT uses BERT's architecture to enable faster, more precise sentence-level comparisons [23].
- LaBSE: A multilingual model based on BERT that supports 109 languages. LaBSE is particularly strong in cross-lingual tasks and performs well even with languages that have limited training data [24].

B. Dataset

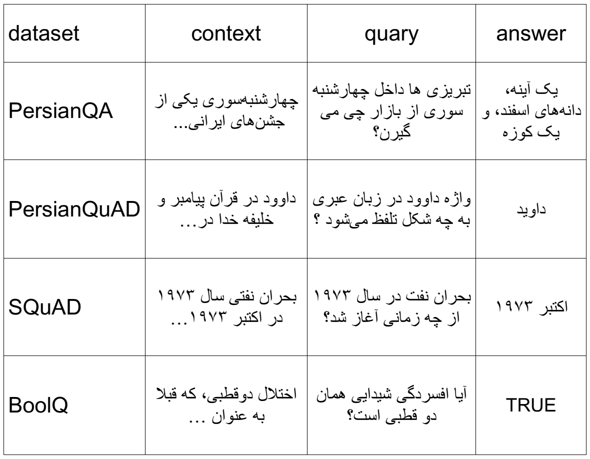
In this study, four different datasets were used to assess and benchmark question-answering models in Persian. Each dataset brings unique features, enabling a thorough evaluation of model performance across various types of questions, passage formats, and answer styles.

The first dataset, PersianQA dataset is a resource for reading comprehension that has been created from articles on Persian Wikipedia. Comprising more than 9,000 entries, this dataset contains both questions that can be answered and those that cannot, which helps improve the model's ability to recognize when information is insufficient for providing a response. Drawing inspiration from the SQuAD2.0 dataset [25], PersianQA incorporates a blend of formal and informal Persian, spanning a wide array of subjects including history and science. One of the standout aspects of this dataset is its complexity, featuring longer passages and diverse types of answers, making it a valuable tool for tasks involving Persian language processing [26].

The second dataset, PersianQuAD, contains approximately 20,000 question-answer pairs, developed from Persian Wikipedia articles. Curated by native Persian speakers, linguistic and cultural accuracy has been emphasized. For evaluation purposes, a subset of 1,000 question-context pairs was utilized, making it suitable for assessing semantic understanding in Persian question-answering systems. A wide variety of question types is included, enhancing the robustness of benchmarking efforts. [27].

The third dataset is a Persian translation of the widely-used SQuAD dataset [28], originally developed in English. Recognized for its large scale with over 107,000 question-answer pairs, the dataset has been translated into Persian using Google Translate to address the need for language-specific resources in Persian natural language processing. The structure of the original dataset has been preserved in the translated version, enabling cross-linguistic comparisons of model performance and providing material for training and testing.

Lastly, a translated version of the BoolQ dataset [29] was utilized, which focuses on yes/no questions. Similar to the SQuAD dataset, BoolQ was translated into Persian using Google Translate. Its uniqueness lies in its emphasis on naturally occurring yes/no questions, derived from anonymized user queries submitted to Google's search engine. This dataset is particularly valuable for evaluating models' abilities to handle binary decision-making tasks in question-answering.



dataset	context	quary	answer
PersianQA	چهارشنبه‌سوری یکی از جشن‌های ایرانی...	تبریزی ها داخل چهارشنبه سوری از بازار چی می گیرن؟	یک آینه، دانه‌های اسفند، و یک کوزه
PersianQuAD	داوود در قرآن پیامبر و خلیفه خدا در…	واژه داوود در زبان عبری به چه شکل تلفظ می‌شود ؟	داوید
SQuAD	بحران نفتی سال ۱۹۷۳ در اکتبر ۱۹۷۳…	بحران نفت در سال ۱۹۷۳ از چه زمانی آغاز شد؟	اکتبر ۱۹۷۳
BoolQ	اختلال دوقطبی، که قبلا به عنوان …	آیا افسردگی شیدایی همان دو قطبی است؟	TRUE

Together, these four datasets offer a comprehensive platform for testing Persian question-answering models, covering a wide range of question formats, passage types, and answer structures.

IV. Methodology

A. Evaluation Process

The Retrieval-Augmented Generation (RAG) model is designed to enhance the handling of question-answering tasks by integrating external information from relevant sources. A crucial aspect of this process involves determining which context is most relevant for each query. The model's performance largely depends on its ability to accurately identify these contexts; successful identification results in more precise answers, thereby improving overall effectiveness. In this section, methods for evaluating and selecting the optimal context for each query are explained.

Three primary methods are employed to measure the similarity between the query and context embeddings:

- L2 Norm (Euclidean Distance): The L2 norm, or Euclidean distance, calculates the straight-line distance between two points in the embedding space. Given two embeddings $\mathbf{q}$ (for the query) and $\mathbf{c}$ (for a context), the L2 norm is given by:

  $$d(\mathbf{q}, \mathbf{c}) = \sqrt{\sum_{i=1}^{n} (q_i - c_i)^2}$$

  where q_i and c_i are the components of the query and context embeddings, respectively, and n is the dimension of the embeddings. A smaller L2 norm value indicates greater similarity, making the context more likely to be relevant.
- Cosine Similarity: Cosine similarity measures the cosine of the angle between the query and context embeddings. It is particularly useful for determining how similar two

vectors are regardless of their magnitude. The cosine similarity is defined as:

$$\text{cosine\_sim}(\mathbf{q}, \mathbf{c}) = \frac{\mathbf{q} \cdot \mathbf{c}}{\|\mathbf{q}\| \|\mathbf{c}\|}$$

where $\mathbf{q} \cdot \mathbf{c}$ is the dot product of the two vectors, and $\|\mathbf{q}\|$ and $\|\mathbf{c}\|$ are their magnitudes. A cosine similarity value closer to 1 indicates that the vectors are highly aligned, implying a stronger relationship between the query and the context.

- Dot Product: The dot product is a fundamental operation in vector algebra that measures the geometric similarity between two vectors. For embeddings $\mathbf{q}$ and $\mathbf{c}$, the dot product is computed as:

$$\mathbf{q} \cdot \mathbf{c} = \sum_{i=1}^{n} q_i c_i$$

A higher dot product between vectors signifies a stronger similarity between the query and the context, as it shows how closely aligned the vectors are in terms of direction. Although the dot product does not consider the magnitudes of the vectors like cosine similarity does, it still provides useful insights into the relevance of the context.

The effectiveness of the RAG model lies in its ability to retrieve relevant contexts and generate accurate responses. By combining similarity measures like L2 norm, cosine similarity, and dot product, the model can effectively identify the best context, enhancing its overall performance. There is potential for further improvement by exploring additional metrics or refining current methods.

This study focuses on comparing different embedding techniques within the RAG framework, with a key emphasis on how well each method retrieves relevant contexts while minimizing retrievals. Fewer retrievals reduce computational load and processing time, making the model more efficient.

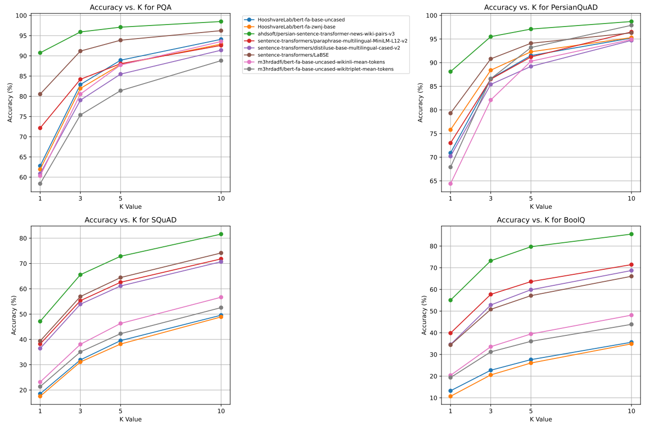
To evaluate the embeddings, the hit@k criterion is used, which measures how effectively relevant contexts are retrieved by an embedding based on the parameter kk. A smaller kk indicates that fewer contexts need to be passed to the LLM, streamlining the process and reducing noise, resulting in more accurate outputs [12].

The embeddings were systematically assessed using hit@k with values of 1, 5, and 10, providing a broad understanding of each embedding's effectiveness within the RAG model.

V. Results and Analysis

In the experiments conducted, 17 different embedding models were tested, and the results of 6 models are provided in the appendix. The complete results are also published on this github page. Among these models, Persian-Sentence-Transformer-News-Wiki-Pairs-v3 emerged as the clear leader. It outperformed the other models across all four datasets, consistently ranking highest on key metrics like hit@1, hit@5, and hit@10. Based on these findings, it can confidently be identified as the top-performing model in the evaluation.

Close behind was LaBSE, Google's multilingual embedding model that supports 108 languages, including Persian. While LaBSE came close in terms of accuracy, it still fell short of Persian-Sentence-Transformer-News-Wiki-Pairs-v3 in overall performance. The other models only began to show competitiveness when the hit@k value was increased beyond 3, which underscores the superior training of the top two models compared to the rest.



Interestingly, little difference was observed between using L2 distance or cosine similarity to evaluate how well the models matched contexts to queries. However, the dot product method was found to be less accurate, although in some cases, its performance was on par with the other two methods.

One challenge noted was that most models struggled with yes/no questions, often failing to link these questions to the correct context. In analyzing results across different hit@k values, it was found that increasing k improved accuracy, even for mid-range models. However, this also increased the amount of context that had to be processed in the next step, which could be considered a limitation.

VI. Conclusion

This paper investigates the role of embedding models in Persian question-answering tasks within the context of the Retrieval-Augmented Generation (RAG) system. Our research indicates that the Persian-Sentence-Transformer-News-Wiki-Pairs-v3 model consistently performs better than other models, such as Google's LaBSE, in retrieving relevant data from the available datasets.

Embeddings serve as a core component in a RAG system by mapping both the input questions and potential answer passages into vectors. These vectors are then compared using similarity measures like cosine similarity or L2 distance to identify the most appropriate content for the model to use when generating a final response.

Integrating powerful embedding models into the RAG pipeline significantly improves both the retrieval and answer generation processes, especially when handling more intricate or nuanced queries. Future research should aim to refine these embeddings to tackle more advanced tasks, such as multi-step reasoning, and to tailor RAG systems for use with low-resource languages like Persian. Enhancing the retrieval phase

TABLE I

Performance of Different Models on Various Datasets by Cosine Similarity(in percentage)

Model	Dataset	Cosine Sim hit@1	Cosine Sim hit@5	Cosine Sim hit@10
HooshvareLab/bert-fa-base-uncased [30]	pqa	62.80	88.92	94.09
HooshvareLab/bert-fa-base-uncased	PersianQuAD	70.90	91.50	95.20
HooshvareLab/bert-fa-base-uncased	SQuAD	18.51	39.52	49.56
HooshvareLab/bert-fa-base-uncased	BoolQ	13.24	27.74	35.41
persian-sentence-transformer-news-wiki-pairs-v3 [31]	pqa	90.75	97.10	98.49
persian-sentence-transformer-news-wiki-pairs-v3	PersianQuAD	88.10	97.10	98.70
persian-sentence-transformer-news-wiki-pairs-v3	SQuAD	47.13	72.86	81.61
persian-sentence-transformer-news-wiki-pairs-v3	BoolQ	55.05	79.45	85.47
paraphrase-multilingual-MiniLM-L12-v2 [32]	pqa	72.15	88.06	92.58
paraphrase-multilingual-MiniLM-L12-v2	PersianQuAD	73.00	91.20	96.50
paraphrase-multilingual-MiniLM-L12-v2	SQuAD	38.18	62.58	71.82
paraphrase-multilingual-MiniLM-L12-v2	BoolQ	39.85	63.52	71.41
distiluse-base-multilingual-cased-v2 [33]	pqa	60.86	85.48	91.40
distiluse-base-multilingual-cased-v2	PersianQuAD	70.20	89.20	94.70
distiluse-base-multilingual-cased-v2	SQuAD	36.44	61.12	70.69
distiluse-base-multilingual-cased-v2	BoolQ	34.65	59.76	68.69
LaBSE [34]	pqa	80.54	93.87	96.24
LaBSE	PersianQuAD	79.30	94.10	96.30
LaBSE	SQuAD	39.38	64.45	74.15
LaBSE	BoolQ	34.40	57.22	65.99
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens [35]	pqa	60.32	87.74	93.55
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens	PersianQuAD	64.4	90.3	94.9
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens	SQuAD	23.14	46.31	56.67
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens	BoolQ	20.33	39.48	48.1

TABLE II

Performance of Different Models on Various Datasets by Dot Product(in percentage)

Model	Dataset	Dot Product hit@1	Dot Product hit@5	Dot Product hit@10
HooshvareLab/bert-fa-base-uncased	pqa	56.67	86.99	93.33
HooshvareLab/bert-fa-base-uncased	PersianQuAD	49.30	80.20	91.20
HooshvareLab/bert-fa-base-uncased	SQuAD	8.34	22.10	30.81
HooshvareLab/bert-fa-base-uncased	BoolQ	4.31	12.60	18.59
persian-sentence-transformer-news-wiki-pairs-v3	pqa	90.54	97.20	98.49
persian-sentence-transformer-news-wiki-pairs-v3	PersianQuAD	88.30	97.30	98.70
persian-sentence-transformer-news-wiki-pairs-v3	SQuAD	46.62	72.66	81.29
persian-sentence-transformer-news-wiki-pairs-v3	BoolQ	54.43	79.24	85.44
paraphrase-multilingual-MiniLM-L12-v2	pqa	67.96	87.96	92.26
paraphrase-multilingual-MiniLM-L12-v2	PersianQuAD	68.10	91.80	96.60
paraphrase-multilingual-MiniLM-L12-v2	SQuAD	33.20	58.82	68.59
paraphrase-multilingual-MiniLM-L12-v2	BoolQ	33.03	58.96	68.17
distiluse-base-multilingual-cased-v2	pqa	63.66	84.62	90.00
distiluse-base-multilingual-cased-v2	PersianQuAD	70.00	89.10	94.90
distiluse-base-multilingual-cased-v2	SQuAD	36.01	60.91	70.12
distiluse-base-multilingual-cased-v2	BoolQ	32.60	59.05	68.26
LaBSE	pqa	80.54	93.87	96.24
LaBSE	PersianQuAD	79.3	94.1	96.3
LaBSE	SQuAD	39.38	64.45	74.15
LaBSE	BoolQ	34.45	57.21	65.99
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens	pqa	56.77	84.52	91.50
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens	PersianQuAD	58.8	89.60	94.50
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens	SQuAD	18.83	40.09	50.80
m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens	BoolQ	18.11	36.76	45.81

through better embedding techniques is key to boosting the overall efficiency of RAG systems, especially in specialized fields like law, academia, or healthcare, where the relevance and accuracy of the information provided are paramount.

References

[1] J. Li, J. Chen, R. Ren, X. Cheng, X. Zhao, J.-Y. Nie, and J.-R. Wen, "The Dawn After the Dark: An Empirical Study on Factuality Hallucination in Large Language Models," in Proceedings of the 2024 ACL, Bangkok, Thailand, 2024, pp. 10879-10899

[2] S. Semnani, V. Yao, H. Zhang, and M. Lam, "WikiChat: Stopping the Hallucination of Large Language Model Chatbots by Few-Shot Grounding on Wikipedia," in Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, Dec. 2023, pp. 2387-2413. doi: 10.18653/v1/2023.findings-emnlp.157

[3] J. Kaddour, J. Harris, M. Mozes, H. Bradley, R. Raileanu, and R. McHardy, "Challenges and Applications of Large Language Models," ArXiv, vol. abs/2307.10169, 2023, doi: Not available. [Online]. Available: https://api.semanticscholar.org/CorpusID:259982665.

[4] M. Douze et al., "The Faiss library," arXiv:2401.08281, 2024.

[5] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Proceedings of the 34th International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 2020, pp. 793-808.

[6] J. F. Ruma, S. Akter, J. J. Laboni, and R. M. Rahman, "A deep learning classification model for Persian Hafez poetry based on the poet's era," Decision Analytics Journal, vol. 4, pp. 100111, 2022, doi: 10.1016/j.dajour.2022.100111.

[7] M. Karrabi, L. Oskooie, M. Bakhtiar, M. Farahani, and R. Monsefi, "Sentiment Analysis of Informal Persian Texts Using Embedding Informal words and Attention-Based LSTM Network," 2020 8th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), 2020, pp. 143-147, doi: 10.1109/CFIS49607.2020.9238699.

[8] M. Esalati, M. J. Dousti, and H. Faili, "Esposito: An English-Persian Scientific Parallel Corpus for Machine Translation," in Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), Torino, Italia, May 2024, pp. 6299-6308, doi: [insert DOI number].

[9] Y. Gao et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," ArXiv, vol. abs/2312.10997, 2023, doi: Not available. [Online]. Available: https://api.semanticscholar.org/CorpusID:266359151

[10] P. Zhang, S. Xiao, Z. Liu, Z. Dou, and J.-Y. Nie, "Retrieve Anything To Augment Large Language Models," arXiv:2310.07554, Oct. 2023.

[11] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking Large Language Models in Retrieval-Augmented Generation," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, pp. 17754-17762, Mar. 2024, doi: 10.1609/aaai.v38i16.29728.

[12] Y. Tang and Y. Yang, "MultiHop-RAG: Benchmarking Retrieval-Augmented Generation for Multi-Hop Queries," in First Conference on Language Modeling, 2024, pp. 1-1. doi: [insert DOI here] Online]. Available: https://openreview.net/forum?id=t4eB3zYWBK

[13] Y. Huang and J. X. Huang, "A Survey on Retrieval-Augmented Text Generation for Large Language Models," ArXiv, vol. abs/2404.10981, 2024. Available: https://api.semanticscholar.org/CorpusID:269188036.

[14] M. S. Zahedi, M. H. Bokaei, F. Shoeleh, M. M. Yadollahi, E. Doostmohammadi, and M. Farhoodi, "Persian Word Embedding Evaluation Benchmarks," in Electrical Engineering (ICEE), Iranian Conference on, 2018, pp. 1583-1588, doi: 10.1109/ICEE.2018.8472549.

[15] D. Jatnika, M. Bijaksana, and A. Ardiyanti, "Word2Vec Model Analysis for Semantic Similarities in English Words," Procedia Computer Science, vol. 157, pp. 160-167, Jan. 2019, doi: 10.1016/j.procs.2019.08.153.

[16] Pennington, Jeffrey, Richard Socher, and Christopher Manning. "Glove: Global Vectors for Word Representation." Proceedings of the Conference on Empirical Methods in Natural Language Processing, vol. 14, 2014, pp. 1532-1542. doi: 10.3115/v1/D14-1162.

[17] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," in Transactions of the Association for Computational Linguistics, vol. 5, Cambridge, MA: MIT Press, 2017, pp. 135-146. doi: 10.1162/tacl-a-00051.

[18] Persian Word Embeddings, "Persian Word Embeddings for Spark NLP," Spark NLP, 2020, [Online]. Available: https://sparknlp.org/2020/12/05/persian_w2v_cc_300d_fa.html. [Accessed: Sept. 10, 2023].

[19] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 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), Minneapolis, Minnesota, 2019, pp. 4171-4186, doi: 10.18653/v1/N19-1423.

[20] M. Farahani, M. Gharachorloo, M. Farahani, and M. Manthouri, "ParsBERT: Transformer-based Model for Persian Language Understanding," Neural Process. Lett., vol. 53, no. 6, pp. 3831-3847, Dec. 2021, doi: 10.1007/s11063-021-10528-4

[21] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv preprint arXiv:1907.11692, 2020, Available: https://openreview.net/forum?id=SyxS0TtvS.

[22] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "MINILM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-trained Transformers," in Proceedings of the 34th International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 2020, pp. 485-498.

[23] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," 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), Hong Kong, China, Nov. 2019, pp. 3982-3992, doi: 10.18653/v1/D19-1410.

[24] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT Sentence Embedding," in Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, May 2022, pp. 878-891, doi: 10.18653/v1/2022.acl-long.62.

[25] Y. Li and Y. Zhang, "Question Answering on SQuAD 2.0 Dataset," Department of Energy Resources Engineering and Department of Electrical Engineering, Stanford University

[26] S. Ayoubi and M. Y. Davoodeh, "PersianQA: a dataset for Persian Question Answering," GitHub, 2021. Accessed: Sep. 12, 2023. [Online]. Available: https://github.com/SajjjadAyobi/PersianQA

[27] A. Kazemi, J. Mozafari, and M. A. Nematbakhsh, "PersianQuAD: The Native Question Answering Dataset for the Persian Language," IEEE Access, vol. 10, pp. 26045-26057, 2022, doi: 10.1109/ACCESS.2022.3157289.

[28] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ Questions for Machine Comprehension of Text," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, 2016, pp. 2383-2392.

[29] C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova, "BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions," in NAACL, 2019.

[30] Hugging Face, HooshvareLab, "ParsBERT (v2.0): A Transformer-based Model for Persian Language Understanding", [Online]. Available: https://huggingface.co/HooshvareLab/bert-fa-base-uncased. Accessed: Sep. 21, 2023.

[31] Hugging Face, Ahdsoft, "Persian Sentence Transformer News Wiki Pairs V3", Hugging Face, 2023, [Online]. Available: https://huggingface.co/ahdsoft/persian-sentence-transformer-news-wiki-pairs-v3. [Accessed: 09/21/2023].

[32] Hugging Face, "Paraphrase-Multilingual-MiniLM-L12-V2," sentence-transformers, 2023. [Online]. Available: https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2. [Accessed: Sep. 22, 2023].

[33] Hugging Face, "sentence-transformers/distiluse-base-multilingual-cased-v2," sentence-transformers, 2023. [Online]. Available: https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v2. [Accessed: Sep. 20, 2023].

[34] Hugging Face, "LaBSE," sentence-transformers, 2023. [Online]. Available: https://huggingface.co/sentence-transformers/LaBSE. [Accessed: Sep. 22, 2023].

[35] Hugging Face, "Bert-fa-base-uncased-wikinli-mean-tokens," sentence-transformers, 2023. [Online]. Available: https://huggingface.co/m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens. [Accessed: Sep. 22, 2023].