Swan: A Family of Arabic-Centric Cross-Lingual Embedding Models

Anonymous ACL submission

Abstract

This paper introduces Swan, a family of cutting-edge embedding models specialized for Arabic language understanding. We present two models, namely Swan-Base and Swan-Large, which are further trained using a large-scale synthetic corpus. To comprehensively evaluate our models, we introduce an extensive text evaluation benchmark, dubbed ArabicMTEB. ArabicMTEB is the largest Arabic text embedding evaluation benchmark to date, covering eight tasks across 74 diverse datasets. Additionally, we propose ArabicMTEBLite, a lightweight and domain-specific synthetic dataset designed for holistic evaluation. Our experiments reveal that Swan-Large exhibits remarkable text embedding capabilities, consistently outperforming all open source models including, Multilingual-E5-large, across all tasks. Furthermore, our efficient model, Swan-Base, also surpasses Multilingual-E5-base in all evaluated tasks. We also explore the impact of synthetic data and the number of hard negatives on the performance of Swan-Base and Swan-Large. Our findings demonstrate that Swan-Base offers an optimal balance between performance, inference time, and cost. Our models will be made publicly accessible for research.

1 Introduction

Natural language processing (NLP) has recently experienced unprecedented growth, prompted by significant breakthroughs in deep learning. Central to these advancements is the development of sophisticated distributed text representations, including word embeddings and sentence embeddings Devlin et al. (2018); Reimers and Gurevych (2019). These embeddings transform sentences into vectors or fixed-length representations, enhancing their utility in various downstream tasks. The prominence of text embeddings, however, extends beyond simple text representation as they are pivotal in enhancing the capabilities of large language models (LLMs) (Touvron et al., 2023b,a; Jiang et al., 2023; Team et al., 2024) within information retrieval systems using retrieval-augmented generation (RAG) (Shao et al., 2023; rag, 2023).In most RAG systems, the information is extracted from a large document using a light embedding model and that information is passed to LLMs like ChatGPT (OpenAI, 2023) GPT4 (OpenAI et al., 2024). Using RAG has shown significant improvements in various question-answering tasks (Lin et al., 2023; rag, 2023) as well as various domain-specific tasks (Bhatia et al., 2024; Shi et al., 2023; Lin et al., 2023)

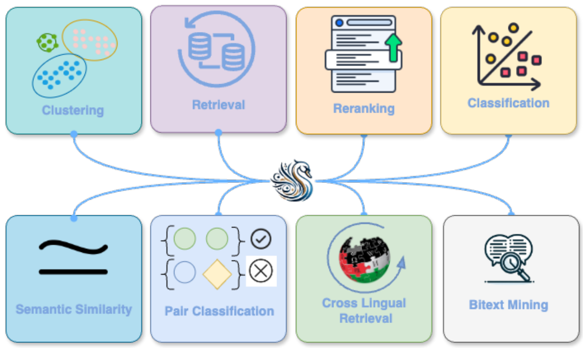


Figure 1: Details of Arabic MTEB

The focus of current embedding models, however, remains primarily on English and Chinese texts, posing substantial limitations when adapting these technologies for other languages and for languages with different scripts. Such limitations are especially pronounced in languages with considerable linguistic divergence from English, such as Arabic, necessitating tailored approaches to develop effective multilingual and language-specific models. This paper explores these themes, focusing on the challenges of extending embedding models to accommodate multilingual contexts and the specific adaptations required for Arabic.

Concretely, we offer a number of contributions: **(1)** We introduce Swan, a family of cutting-edge embedding for Arabic. We propose two models: Swan-Base, based on ARBERTv2 (Elmadany et al., 2022) and Swan-Large, based on ArMistral-chat, an in-house further trained SoTA Arabic LLM that we further trained using a large synthetic corpus generated using Cohere Command R+[1] model. We also introduce **(2)** ArabicMTEB, an extensive and massive text evaluation benchmark. ArabicMTEB is the largest Arabic text embedding evaluation benchmark and the only one that measures cross-lingual retrieval for Arabic as one language, encompassing eight tasks across 74 datasets. **(3)** We introduce ArabicMTEBLite a lightweight domain-specific synthetic dataset for holistic evaluation of models on various Arabic domains. **(4)** Our large model, Swan-Large, demonstrates exceptional text embedding capabilities, achieving SoTA performance by outperforming Multilingual-E5-large (Wang et al., 2024b) in *all* Arabic tasks. Moreover, our efficient model, Swan-Base, surpasses Multilingual-E5-base (Wang et al., 2024b) in *all* Arabic tasks. **(5)** We also explore the impact of synthetic data and the number of hard negatives on Swan-Base and Swan-Large, demonstrating that Swan-Base is optimized for *latency* and *performance*.

The rest of the paper is organized as follows: In Section 2, we review related work with a particular emphasis on Arabic text embedding models, their applications and challenges. Section 3 outlines how we built our benchmark dataset, ArabicMTEB. We present our approach to model training Swan models in Section 4. Section 5 is about our experiments and model analysis. We discuss our results in Section 6, including the impact of using synthetic data and the number of hard negatives, as well as model latency. Finally, we conclude in Section 7.

2 Related Works

Multilingual text embedding models are essential for enabling cross-lingual understanding and retrieval tasks. Recent models such as the M3-Embedding (Chen et al., 2024) can handle multiple languages, functions, and input granularities. Similarly, Wang et al. (2024b) present the Multilingual E5 Text Embeddings, which leverage large-scale multilingual data for training embeddings efficiently in various languages. These developments indicate a strong trend towards models that are not only efficient but also versatile across linguistic contexts. Additionally, the Gecko model (Lee et al., 2024) illustrates the benefits of knowledge distillation from LLMs into a more compact embedding model that retains high retrieval performance across languages.

Text Embedding Benchmarks play a pivotal role in measuring the progress and effectiveness of text embedding models. The Massive Text Embedding Benchmark (MTEB) (Muennighoff et al., 2022) provides a vast framework for evaluating different embedding approaches across a wide array of tasks and languages. Xiao et al. (2023) propose a new Chinese Massive text embedding benchmark (C-MTEB) focused on specific Chinese tasks. Similarly, Wehrli et al. (2024); Mohr et al. (2024) propose benchmarks for German and Spanish text embeddings, highlighting the specific requirements of language-focused evaluations.

Arabic Embeddings and Benchmarks. Specific efforts have been made towards developing and benchmarking Arabic language models and embeddings. Abdul-Mageed et al. (2020) introduce ARBERT and MARBERT, deep bidirectional transformers specifically aimed at a multi-dialectal Arabic understanding. These models have set new standards in Arabic by addressing the unique challenges of Arabic varieties. On the benchmarking front, Elmadany et al. (2022) present ORCA, a comprehensive Arabic language understanding benchmark that includes multiple datasets and tasks to cover the diversity of Arabic. Furthermore, the Dolphin benchmark (Nagoudi et al., 2023) focuses on Arabic language generation, providing a broad range of tasks to assess the generative capabilities of Arabic models. These initiatives contribute to the field by providing tailored resources and benchmarks that enhance the development of Arabic-specific models.

In summary, the works reviewed here collectively shape the evolving landscape of text embeddings, providing insights that can further impact the development of Arabic text embedding models. To our knowledge, our work is the first to focus on Arabic text embedding models, benchmarks, and crosslingual retrieval in one full swoop.

3 ArabicMTEB Benchmark

In this work, we introduce ArabicMTEB, a comprehensive benchmark specifically designed for evaluating the generality of Arabic text embeddings (Figure 1). Recent years have seen the development of

[1] https://docs.cohere.com/docs/command-r-plus

Task	Datasets	Languages	Dialects
ArRTR	15	1	4
CRTR	12	6	-
CLF	18	1	6
BTM	12	5	8
RRK	5	2	-
STS	5	1	-
CLR	4	1	-
PairCLF	3	1	-
Total*	**74**	**11**	**9**

Table 1: Overview of our Datasets. **ArRTR**: Arabic Retrieval, **STS**: Semantic Textual Similarity, **PairCLF**: Pair Classification, **CLF**: Classification, **CLR**: Clustering, **RRK**: Reranking, **BTM**: BiTextMining, **CRTR**: Crosslingual Retrieval. *Total represents the unique languages.

pivotal datasets for studying Arabic NLP, such as ORCA (Elmadany et al., 2023), Dolphin (Nagoudi et al., 2023), and MTEB (Muennighoff et al., 2022). None of these works, however, has focused on specific aspects of Arabic text embeddings models. For this work, we curated 74 datasets for evaluating Arabic text embeddings. We group the datasets based on the capabilities of the embeddings they assess. More specifically, we cover the following categories: *retrieval*, *re-ranking*, *semantic textual similarity*, *classification*, *pair classification*, and *clustering*. Each category, drawing datasets from varied domains, comprehensively evaluates a specific capability of the embeddings. An overview of the datasets is in Table 1 and Table 2.

One central area of focus is the cross-lingual transfer of information, and we have specifically focused on cross-lingual reranking and retrieval tasks in Arabic and six other languages: *English, German, Spanish, Chinese, Vietnamese,* and *Hindi*. As seen from Table 2, our *ArabicMTEB* is the only benchmark to include Arabic and the largest and most comprehensive benchmark.

3.1 Tasks and Evaluation Datasets

In ArabicMTEB, we assess the capabilities of embeddings through various tasks using specific datasets. Each dataset is tailored to evaluate different aspects of embedding performance in real-world conditions as we explain next.

Arabic Retrieval. This task involves using test queries to find Top-k similar documents in a large corpus. We adopt BEIR's (Thakur et al., 2021) methodology, primarily using NDCG@10 as the metric. Here we have 15 different datasets which are long form question-answering datasets from (Nagoudi et al., 2023). We include dialects from Saudi Arabia, Egypt, Yemen and Jordan, along with MSA. Other datasets include MLDR (Chen et al., 2024) and XPDA (Shen et al., 2023), which measure how well embeddings identify top-relevant documents from large corpora that include Arabic.

Bitext Mining. This task requires matching sentences from two different language collections to identify translations, focusing on dialects and language pairs such as Moroccan to French and Arabizi to English. Datasets for evaluation are taken from Nagoudi et al. (2023). These datasets are originally for code switched machine translation but we adapt them for bitext mining, using cosine similarity to score sentence pair matches. Here we have dialects from Algeria, Egypt, Jordan, Lebanon, Moroccan, MSA, Saudi Arabia, and Yemen. Our bitext mining collection comprises 12 datasets in total.

Cross-Lingual Retrieval. Using Arabic queries to find Top-k similar documents in a corpus in a different language, this task uses the Mmarco Dev set (Bonifacio et al., 2021a), which spans several language pairs from Arabic and six other languages: *English, German, Spanish, Chinese, Vietnamese*, and *Hindi*.

Re-Ranking. Candidate documents for test queries are re-ranked based on embedding similarity. Datasets such as the MIRACL (Zhang et al., 2022a), which offers a multilingual perspective with an emphasis on Arabic and English, test the ability of embeddings to reorder documents effectively. Here we have five datasets in total.

Semantic Textual Similarity (STS). This task measures the correlation between the embeddings of two sentences. We follow the protocol from Sentence-BERT (Reimers and Gurevych, 2019), primarily using Spearman's correlation. Datasets like STS17 and STS22 (Cer et al., 2017b) evaluate how well embeddings capture the semantic nuances between sentences.We employ five datasets in this category.

Classification. This task utilizes embeddings to predict labels from input data, using datasets like ORCA (Elmadany et al., 2022), which covers Arabic classification, including six different dialects, assessing the ability to categorize text into predefined labels. This is our largest task with 18 multi domain multi dialectal datasets.

Pair-classification. Predicting the relationship be-

Benchmark	Language	Tasks	Datasets	#Tasks	CRTR	Arabic
MTEB	English	*RTR, STS, PairCLF, CLF, RRK, CLR, SUM*	56	7	×	✓
C-MTEB	Chinese	*RTR, STS, PairCLF, CLF, RRK, CLR*	35	6	×	×
De-MTEB	German	*RTR, STS, PairCLF, CLF, RRK, CLR*	17	6	×	×
F-MTEB	French	*RTR, STS, PairCLF, CLF, RRK, CLR, BTM*	17	7	×	×
Es-MTEB	Spanish	*RTR, STS, PairCLF, CLF, RRK, CLR*	17	6	×	×
Polish-MTEB	Polish	*RTR, STS, PairCLF, CLF, CLR*	26	5	×	×
	Danish				×	×
Scand. MTEB	Norwegian	*RTR, CLF, BTM, CLR*	26	4	×	×
	Swedish				×	×
ArabicMTEB	Arabic	*ArRTR, STS, PairCLF, CLF, RRK, CLR, BTM, CRTR*	74	8	✓	✓

Table 2: Comparison of Massive Text Embedding benchmarks proposed in the literature across the different covered task clusters. **RTR**: Retrieval, **ArRTR**: Arabic Retrieval, **STS**: Semantic Textual Similarity, **PairCLF**: Pair Classification, **CLF**: Classification, **CLR**: Clustering, **RRK**: Reranking, **BTM**: BitextMining, **CRTR**: Crosslingual Retrieval.

tween a pair of sentences using embedding similarity is tested using datasets such as XNLI (Conneau et al., 2018) and PairCLF (Cer et al., 2017b), focusing on understanding relationships between sentence pairs. Here use three datasets in this category.

Clustering. Grouping sentences into clusters using mini-batch k-means, this task uses datasets like Arabic News Articles which are collected from Al-Jazeera and Baly et al. (2018a) *stance headings*, which evaluate the effectiveness of embeddings in clustering related content. Here we have four datasets. Each dataset in ArabicMTEBis meticulously chosen to cover a broad spectrum of linguistic and semantic scenarios, ensuring a comprehensive evaluation of Arabic text embeddings.

3.2 ArabicMTEBLite Benchmark

Due to the large size of the ArabicMTEB, it is not feasible to evaluate proprietary embedding models. Therefore, we have developed a novel benchmark to address the need for robust domain-specific models in Arabic information retrieval, specializing in domains such as news, finance, legal, medical, and general knowledge. This benchmark is light and easy to run, yet we believe it represents the closest evaluation to real-time scenarios. Creation of this benchmark involved the conversion of Arabic documents from these as well as Wikipedia. We split and chunk the documents into texts of 1,024 lengths. We then randomly select chunks and ask GPT4-Turbo (OpenAI et al., 2024) to generate five different styles of queries for each chunk. Consequently, we filter out duplicate and repeated queries using GPT4-Omni (OpenAI et al., 2024) to ensure a high-quality evaluation dataset. ArabicMTEB contains a total of $10k$ queries and $100k$ documents from various domains as described above.

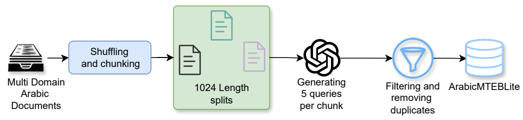


Figure 2: Generation pipeline for our ArabicMTEBLite Benchmark.

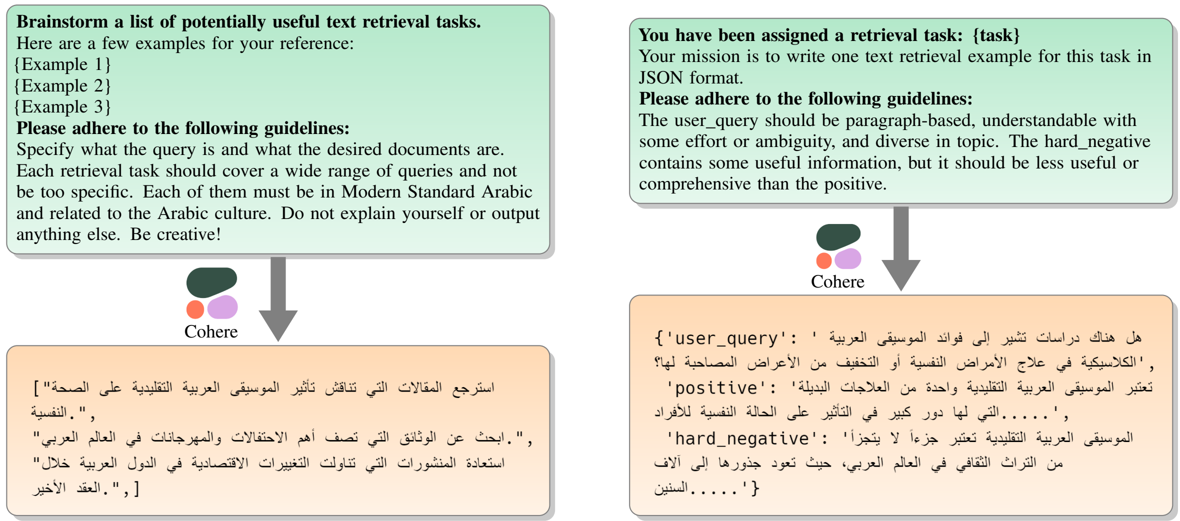
Family	Language	Dataset	Type	Size
Monolingual	Arabic	Synthetic	Paragraph	100K
		ORCA MMARCO-ar	Sentence	500K 8.1M
Crosslingual	Arabic to 15 Langs Arabic to 6 Langs	MMARCO XOR-TyDi	Sentence	3M 20.5K
Multilingual	11 Langs 16 Langs	Mr-Tydi Miracl	Sentence	49K 343K
Total				12.3M

Table 3: The diverse datasets employed for training our embedding language models.

4 Swan

4.1 Training Data

We develop the largest training corpus for Arabic embedding models, leveraging a unique assembly of datasets to ensure comprehensive linguistic coverage and diversity. Our training strategy employs paragraph-based and sentence-based datasets, meticulously curated from multiple sources, enhancing the model's ability to effectively understand Arabic text. Table 3 shows an overview of our training datasets. The datasets can be categorized into three main categories: Arabic-specific, cross-lingual, and multilingual. The *Arabic-specific* datasets focus on enhancing the model's performance in handling various forms of Arabic text. *Cross-lingual* datasets, particularly those facilitat-



(a) Query Generation

(b) Positive and hard negative generation

Figure 3: Methodology to generate our synthetic data.

ing translation between Arabic and 15 other languages, are crucial for applications involving multiple languages. Finally, the *multilingual* datasets incorporate data from multiple languages, further enriching the model's capability to operate in a global multilingual environment.

Arabic Datasets. We use two primary sources of data: ORCA (Elmadany et al., 2023) and mMARCO-ar (Bonifacio et al., 2021a). ORCA is a compilation of labelled datasets with multiple tasks such as semantic text similarity (STS), sentence classification, text classification, natural language inference (NLI), and question answering. We use all the training sets from ORCA, encompassing 60 different datasets. These datasets are used as the Arabic monolingual data after cleaning up and de-duplication using the pipeline developed by Bhatia (2023), which is further described in Appendix D. The de-duplication process removes data with a lot of noise. Additionally, we generate a $100k$ paragraph-to-paragraph synthetic dataset using the Cohere Command R+ model, which is proficient in generating Arabic texts. We used the same method as Wang et al. (2023), utilizing a large Arabic text dataset comprising 100M documents as seed data. This multi-domain seed data focuses on various areas such as news, finance, medicine, and legal text. The data generation process used four A100 GPUs and vLLM (Kwon et al., 2023) as the inference accelerator. The format of the prompts used to instruct the Cohere Command R+ model can be found in Figure 3.

Cross-Lingual Dataset. The mMARCO dataset comprises translations of the MS MARCO dataset into 15 languages (Bonifacio et al., 2021b). To ensure that documents correspond accurately to their queries in different languages, we utilize specific IDs. We create $100k$ samples for each cross-lingual pair and shuffle the IDs to prevent repetition, thus guaranteeing that unique data samples are employed for each language.

Multilingual Datasets. We utilize the MIRACL (Zhang et al., 2022b) and Mr.TyDi (Zhang et al., 2021) datasets as our multilingual resources to enhance our model's capability in understanding multiple languages, ensuring it performs effectively across various multilingual tasks.

4.2 Hard-Negatives Selection

To enhance the model's accuracy, it is crucial to use negative documents closely aligned with the query's context (Karpukhin et al., 2020). This is achieved by leveraging advanced models such as the multilingual-E5 models from Wang et al. (2024b). The process involves converting all documents into a vector form within the embedding space. Subsequently, these document embeddings are compared using the cosine similarity score to establish their relevance to the query. Once all documents are scored, they are sorted by their similarity to the query. The top-ranked document is typically the positive example, while the rest are potential negatives. To rigorously test the model's performance with varying degrees of difficulty, we systematically select negative samples in increasing batch sizes—specifically, batches from the set *{1, 3, 7, 15, 31}*. This method allows us to observe the impact of introducing more challenging or "hard" negatives into the training process. We only generate hard negatives for the Arabic subset of our training data from Section 4.1.

4.3 Training Strategy

Our training recipe is inspired by RankLlama (Ma et al., 2023) and the BGE models (Xiao et al., 2023). We use LoRA (Hu et al., 2021) for our large model's parameters and full training for the small model. We train our models for three epochs on the entire dataset, using a learning rate of $5e^{-6}$ and a constant batch size of 128. To optimize performance, we included seven hard negatives in the training process. Further details of the training process can be found in Appendix B.

4.4 Evaluation

We evaluate our trained model on our Arabic massive text embedding benchmark, ArabicMTEB (section 3), based on MTEB (Muennighoff et al., 2022), with enhanced settings for improved Arabic understanding. Evaluation is conducted using prompts from Table 10, on both ArabicMTEB and ArabicMTEBLite. For document retrieval tasks, we use NDCG@10 to measure retrieval quality. Bitext Mining employs the F_1 score for sentence pair alignment. Re-ranking of documents uses the MAP score for ordering candidate documents. Semantic Textual Similarity (STS) uses *Spearman's correlation* for semantic similarity, while Classification and Pair-classification tasks use average *precision*. Clustering employs the *V-measure* score to assess cluster coherence.

5 Experiments

This paper introduces two models, Swan-Base built with ARBERTv2 (Abdul-Mageed et al., 2021a) and Swan-Large based on an in-house further pretrained Mistral-7B model(Jiang et al., 2023), dubbed ArMistral-7B. As seen from Elmadany et al. (2022) ARBERTv2 is a powerful SoTA Arabic NLU model pretrained on a $30B$ tokens dataset. We further pretrain Mistral-7B using a $35B$ tokens large corpus of Arabic text datasets which we clean, filtered and de-duplicate using an in-house pre-processing pipeline as described in Appendix D. We then instruction finetune the model using a large dataset of instructions from Huang et al. (2024) and align it using DPO and SimPO (Rafailov et al., 2023; Meng et al., 2024). This model is a top-performing model in all Arabic generation tasks, and we have shared our in-house results in Appendix A. We also compare the performance of our models to 12 other baseline models. We evaluated with two versions of MARBERT (Abdul-Mageed et al., 2020), two versions of ARBERT (Abdul-Mageed et al., 2021b), two versions of ARBERTv2 (Elmadany et al., 2022), four versions of CamelBERT (Inoue et al., 2021) and four versions of the multilingual E5 models (Wang et al., 2024b,a).

5.1 ArabicMTEB Results

We present the results of our evaluation on all tasks in Table 4.

Swan-Base. With a smaller size of 164M parameters, Swan-Base shows strong capabilities, particularly in classification, where it outperforms all other models with a score of 57.34. This model also performs robustly in Pair classification (74.93) and achieves a respectable average of 57.21. Since Swan-Base is based on ARBERTv2, which performs well on classification tasks, our model further improves the results on ARBERTv2 scores.

Swan-Large. Swan-Large, 7.23B parameters, outperforms all other models in most of the evaluated tasks. It scores highest in Retrieval (65.63), Pair classification (75.62), and Bitext mining (71.24), with an impressive average score of 62.11. Its performance in STS is also noteworthy, achieving a close second-highest score (59.10), marginally below the best-performing model in this category. This strong performance shows the efficacy of our training data as well as our use of a larger LLM based on the ArMistral-7B, which has been extensively trained on a diverse Arabic dataset.

The comparison also includes several versions of well-known Arabic encoder models such as MARBERT, ARBERT, ARBERT-v02, CamelBERT, and the multilingual E5 series as seen in Table 11. Notably, the multilingual-e5-large model emerges as a strong overall model, securing the second-best average score (61.65) and excelling in STS (59.45) and Re-ranking (70.79).

5.2 ArabicMTEBLite Results

We compare the Swan models with proprietary models by OpenAI and Cohere. These two are considered the SoTA in the area of embedding models. As seen from Table 5 Swan-Large per-

Model	Size	Dim.	RTR	STS	PairCLF	CLF	RRK	CLR	BTM	Avg.
ARBERTv2	164M	768	15.12	37.88	62.87	56.85	62.21	39.25	1.99	39.45
text2vec-base-multilingual	118M	384	27.69	**59.37**	71.41	47.94	57.76	37.26	38.32	48.54
LaBSE	471M	768	34.98	54.15	70.60	49.57	62.17	41.42	33.28	49.45
multilingual-e5-small	118M	384	55.14	56.73	73.97	50.85	67.92	42.37	38.47	55.06
multilingual-e5-base	278M	768	56.91	57.99	74.30	52.30	**69.07**	**42.56**	33.90	55.29
Swan-Small	164M	768	**58.42**	58.44	74.93	**57.34**	68.43	40.43	**42.45**	**57.21**
e5-mistral-7b-instruct	7.11B	4096	56.34	57.02	70.24	53.21	66.24	39.44	70.50	59.00
multilingual-e5-large	560M	1024	64.01	**59.45**	75.06	**53.43**	**70.79**	**42.49**	66.33	61.65
Swan-Large	7.23B	4096	**65.63**	59.10	**75.62**	52.55	69.42	41.24	**71.24**	**62.11**

Table 4: ArabicMTEBResults Here we compare our models in two different classes small and large. **ArRTR**: Arabic Retrieval, **STS**: Semantic Textual Similarity, **PairCLF**: Pair Classification, **CLF**: Classification, **CLR**: Clustering, **RRK**: Reranking, **BTM**: BiTextMining, **CRTR**: Crosslingual Retrieval.

Model	News	Legal	Medical	Finance	Wikipedia	Avg	Cost
Openai-3-large	88.10	**89.68**	**80.24**	**61.46**	91.52	**82.20**	3.88$
Swan-Large	**90.42**	87.90	79.64	55.34	**93.10**	81.28	0.75$
Cohere-v3.0	85.23	86.52	63.27	42.80	90.96	73.76	1.54$
Swan-Base	81.55	78.86	70.97	42.48	80.46	70.86	**0.44$**
Openai-3-small	71.42	85.23	71.50	32.90	82.20	68.65	1.75$
Cohere-light-v3.0	70.32	86.83	67.68	22.68	90.34	67.57	0.55$
Openai-ada-002	65.34	81.83	71.76	39.62	76.79	67.07	1.66$

Table 5: ArabicMTEBLite Results.

Model (HN)	1	3	7	15	31
Swan-Base	48.84	52.19	54.13	**56.25**	51.93
Swan-Large	59.48	59.35	**60.42**	59.44	59.83

Table 6: Impact of number of Hard Negatives (HN).

forms competitively with text-embedding-3-large (with an average score of 81.28 for Swan-large compared to 82.20 for text-embedding-3-large). We also see that Swan-Large outperforms embed-multilingual-v3.0 by Cohere, a very strong multilingual model. Our Swan-Base outperforms text-embedding-3-small, text-embedding-ada-002 by OpenAI and embed-multilingual-light-v3.0 by Cohere in terms of performance on ArabicMTEBLite. Table 5 also shows that models struggle to find the right documents in the financial domain, suggesting further scope for improvement through building domain-specific models (Bhatia et al., 2024).

In addition, we show the cost of evaluating these models on ArabicMTEBLite, which contains $10k$ queries and $100k$ documents using the OpenAI and Cohere APIs. We evaluate Swan models on a single V100 32 GB GPU, which costs 2.30$ an hour. As Table 5 shows, our models are the *most economical* in the entire range and have very strong performance. When comparing the performance-cost trade-of, our models emerge as much better suited than OpenAI and Cohere models.

6 Discussion

In this section, we explore the effects of (i) incorporating synthetic data and (ii) varying the number of hard negatives on our models. We also evaluate and compare the latency of all the models.

Impact of Hard Negatives: Hard negatives are challenging examples that are nearly correct but ultimately incorrect, forcing the model to learn more nuanced distinctions between the different classes. Our experiments focuse on assessing the impact of varying the hard negatives used while training our models, Swan-Large and Swan-Base. We train each model with different quantities of hard negatives. Namely, we experiment with using 1, 3, 7, 15, and 31 hard negatives per training instance.

Swan-Large show a peak in performance with 60.42 when trained with seven hard negatives, indicating an optimal level of challenge that enhances learning without overwhelming the model. Interestingly, further increases in hard negatives does not improve performance, suggesting a threshold beyond which additional complexity does not translate to better learning outcomes.

Swan-Base reaches its highest performance at 56.25 with 15 hard negatives. This model shows a general upward trend in performance as the number of hard negatives increases, peaking at 15, but then declining slightly when the number is increased to 31. This pattern suggests that while additional hard negatives initially provide beneficial learning challenges, there can be a point of diminishing returns where too much complexity hinders further learning.

Impact of Synthetic Data. Synthetic data has become increasingly popular in training machine learning models, particularly when real-world data is scarce or lacks diversity. This approach aims to enhance the models' ability to generalize across

Model	RTR	STS	PairCLF	CLF	RRK	CLK	BTM	Avg.
Swan-Base	15.12	37.88	62.87	**56.85**	62.21	39.25	1.99	39.45
+ Arabic	28.39	41.49	70.25	51.89	68.57	39.12	**18.74**	45.49
+ Synthetic	**31.07**	**55.78**	**74.23**	54.27	**68.88**	**39.43**	18.19	**48.84**
Swan-Large	44.46	48.63	72.34	50.43	69.39	38.28	44.2	52.53
+ Arabic	54.53	52.93	75.24	**52.54**	70.49	40.21	48.35	56.33
+ Synthetic	**56.34**	**57.89**	**76.90**	50.21	**70.92**	**41.76**	**62.34**	**59.48**

Table 7: Impact of using Synthetic data.

different contexts and improve their robustness against unusual or rare linguistic patterns. As shown in Table 7, the incorporation of synthetic data impacts the performance of both models across all tasks. For the Swan-Base model, adding synthetic data resulted in substantial improvements in several key performance metrics: Retrieval saw an increase from 15.12 to 31.07, Semantic Textual Similarity jumped from 37.88 to 55.78, and Pair Classification from 62.87 to 74.23. The notable boost in STS is particularly significant, suggesting that the synthetic data helps the model better understand and process complex semantic relationships within texts. For the Swan-Large model, the results are similarly encouraging. The model performs better across all evaluated tasks when trained with synthetic data. For instance, the score in Bitext Mining soared from 44.20 to 62.34, highlighting a major improvement in the model's capability to identify and align text pairs across languages, an essential task for evaluating the quality of machine translation. Moreover, synthetic data helped to elevate the model's performance in STS from 48.63 to 57.89 and in Pair classification from 72.34 to 76.90.

Inference Latency. Inference latency is very critical in deploying machine learning models, especially in real-time applications with crucial response time. It refers to the time taken by a model to predict received input. In the context of text embedding models such as Swan-Base and Swan-Large, lower latency is particularly valuable for user-facing services that rely on fast processing of natural language input, such as chatbots and search engines. From Figure 4, we find that Swan-Large, despite its larger size indicated by a larger bubble, has optimized inference times due to architectural efficiencies, and Swan-Base strikes the perfect balance between size, performance, and latency. We compare the performance of the models from Table 4.

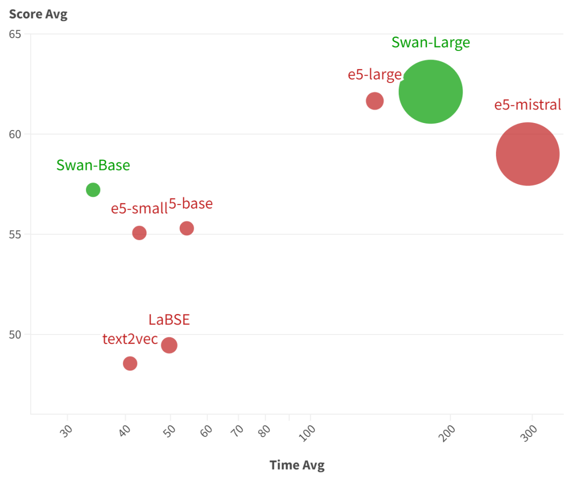


Figure 4: Latency vs Performance.

7 Conclusion

In this paper, we introduced Swan-Large and Swan-Base, along with the comprehensive ArabicMTEB benchmark for evaluating Arabic text embeddings. Our models demonstrate outstanding performance, benefiting from the strategic use of hard negatives and synthetic data in training. These approaches enhance model robustness and generalization capabilities, essential for handling complex linguistic scenarios. Additionally, our models achieves efficient inference times, making them suitable for real-time applications. These results set new benchmarks in Arabic text embeddings, paving the way for future advancements in multilingual text analysis.

8 Limitations

While the development of the Amwaj models and the introduction of the ArabicMTEB benchmark mark significant advancements in Arabic text embeddings, there are some limitations to consider:

- **Synthetic Data Dependency**: The reliance on synthetic data for training and evaluation, while beneficial in some respects, introduces potential biases and does not fully capture the diversity and complexity of real-world data. This could lead to models that perform well on synthetic benchmarks but may not generalize as effectively in real-world applications.

- **Cross-Lingual Performance**: Although the Amwaj models demonstrate strong performance in cross-lingual tasks, the evaluation is primarily focused on a limited set of language pairs. The generalizability of these results to a broader range of languages, especially low-resource languages, remains uncertain.

- **Dialectal Variations**: Arabic is a highly dialectal language, and while the models incorporate multiple dialects, the coverage and performance across all major dialects are not uniformly robust. This could affect the usability of the models in regions where certain dialects predominate.

- **Inference Latency**: Despite optimizations, the larger model, Amwaj-Large, still presents higher inference latency, which could be a barrier to real-time applications. The trade-off between model size, performance, and latency needs further exploration to enhance practicality.

- **Ethical and Bias Concerns**: The use of synthetic data and the inherent biases in training corpora raise ethical concerns about fairness and representation. The models might inadvertently perpetuate or amplify existing biases in the data, which warrants careful consideration and mitigation strategies.

9 Ethical Statement

All research and development activities for the Swan models and ArabicMTEB benchmark were conducted with a commitment to ethical standards. Data collection and usage adhered to privacy and confidentiality norms, ensuring no sensitive information was utilized without proper anonymization and consent. We acknowledge the potential biases introduced by synthetic data and have taken steps to mitigate these through diverse data sources and rigorous evaluation.

References

2023. Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering. *Transactions of the Association for Computational Linguistics*.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2020. Arbert & marbert: Deep bidirectional transformers for arabic. ACL-2021 camera ready version.

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

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

Manel Aloui, Hasna Chouikhi, Ghaith Chaabane, Haithem Kchaou, and Chehir Dhaouadi. 2024. 101 billion arabic words dataset. *Preprint*, arXiv:2405.01590.

Ramy Baly, Mitra Mohtarami, James Glass, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. 2018a. Integrating stance detection and fact checking in a unified corpus. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 21–27, New Orleans, Louisiana. Association for Computational Linguistics.

Ramy Baly, Mitra Mohtarami, James Glass, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. 2018b. Integrating stance detection and fact checking in a unified corpus. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 21–27.

Gagan Bhatia. 2023. PolyDeDupe.

Gagan Bhatia, El Moatez Billah Nagoudi, Hasan Cavusoglu, and Muhammad Abdul-Mageed. 2024. Fintral: A family of gpt-4 level multimodal financial large language models. *Preprint*, arXiv:2402.10986.

Luiz Bonifacio, Vitor Jeronymo, Hugo Queiroz Abonizio, Israel Campiotti, Marzieh Fadaee, Roberto Lotufo, and Rodrigo Nogueira. 2021a. mmarco: A multilingual version of the ms marco passage ranking dataset.

Luiz Bonifacio, Vitor Jeronymo, Hugo Queiroz Abonizio, Israel Campiotti, Marzieh Fadaee, Roberto Lotufo, and Rodrigo Nogueira. 2021b. mmarco: A multilingual version of the ms marco passage ranking dataset.

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. 2017a. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017b. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Zhihong Chen, Shuo Yan, Juhao Liang, Feng Jiang, Xiangbo Wu, Fei Yu, Guiming Hardy Chen, Junying Chen, Hongbo Zhang, Li Jianquan, Wan Xiang, and Benyou Wang. 2023. MultilingualSIFT: Multilingual Supervised Instruction Fine-tuning.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ibrahim Abu El-Khair. 2016. 1.5 Billion Words Arabic Corpus. *arXiv preprint arXiv:1611.04033*.

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

AbdelRahim Elmadany, ElMoatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. ORCA: A challenging benchmark for Arabic language understanding. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9559–9586, Toronto, Canada. Association for Computational Linguistics.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, et al. 2022. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages. *arXiv preprint arXiv:2204.08582*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Huang Huang, Fei Yu, Jianqing Zhu, Xuening Sun, Hao Cheng, Dingjie Song, Zhihong Chen, Abdulmohsen Alharthi, Bang An, Juncai He, Ziche Liu, Zhiyi Zhang, Junying Chen, Jianquan Li, Benyou Wang, Lian Zhang, Ruoyu Sun, Xiang Wan, Haizhou Li, and Jinchao Xu. 2024. Acegpt, localizing large language models in arabic. *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. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, Kyiv, Ukraine (Online). Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Jinhyuk Lee, Zhuyun Dai, Xiaoqi Ren, Blair Chen, Daniel Cer, Jeremy R. Cole, Kai Hui, Michael Boratko, Rajvi Kapadia, Wen Ding, Yi Luan, Sai Meher Karthik Duddu, Gustavo Hernandez Abrego, Weiqiang Shi, Nithi Gupta, Aditya Kusupati, Prateek Jain, Siddhartha Reddy Jonnalagadda, Ming-Wei Chang, and Iftekhar Naim. 2024. Gecko: Versatile text embeddings distilled from large language models.

Weizhe Lin, Rexhina Blloshmi, Bill Byrne, Adria de Gispert, and Gonzalo Iglesias. 2023. Li-rage: Late interaction retrieval augmented generation with explicit signals for open-domain table question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*

(*Volume 2: Short Papers*), pages 1557–1566, Toronto, Canada. Association for Computational Linguistics.

Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2023. Fine-tuning llama for multi-stage text retrieval. *Preprint*, arXiv:2310.08319.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *Preprint*, arXiv:2405.14734.

Isabelle Mohr, Markus Krimmel, Saba Sturua, Mohammad Kalim Akram, Andreas Koukounas, Michael Günther, Georgios Mastrapas, Vinit Ravishankar, Joan Fontanals Martínez, Feng Wang, Qi Liu, Ziniu Yu, Jie Fu, Saahil Ognawala, Susana Guzman, Bo Wang, Maximilian Werk, Nan Wang, and Han Xiao. 2024. Multi-task contrastive learning for 8192-token bilingual text embeddings. *Preprint*, arXiv:2402.17016.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.

El Moatez Billah Nagoudi, AbdelRahim Elmadany, Muhammad Abdul-Mageed, and Tariq Alhindi. 2020. Machine generation and detection of Arabic manipulated and fake news. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 69–84, Barcelona, Spain (Online). Association for Computational Linguistics.

El Moatez Billah Nagoudi, AbdelRahim Elmadany, Ahmed El-Shangiti, and Muhammad Abdul-Mageed. 2023. Dolphin: A challenging and diverse benchmark for arabic nlg. *Preprint*, arXiv:2305.14989.

OpenAI. 2023. Chatgpt: Optimizing language models for dialogue. *OpenAI*. https://openai.com/research/chatgpt.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, ukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, ukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Preprint*, arXiv:2305.18290.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Singapore. Association for Computational Linguistics.

Xiaoyu Shen, Akari Asai, Bill Byrne, and Adrià de Gispert. 2023. xpqa: Cross-lingual product question answering across 12 languages. *Preprint*, arXiv:2305.09249.

Feng Shi, Ruifeng Ren, Xiaoying Feng, and Wenjie Li. 2023. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructure. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*. Leibniz-Institut für Deutsche Sprache.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. 2024. Gemma: Open models based on gemini research and technology. *Preprint*, arXiv:2403.08295.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023a. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *Preprint*, arXiv:1807.03748.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2023. Improving text embeddings with large language models.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024a. Improving text embeddings with large language models. *Preprint*, arXiv:2401.00368.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024b. Multilingual e5 text embeddings: A technical report.

Silvan Wehrli, Bert Arnrich, and Christopher Irrgang. 2024. German text embedding clustering benchmark.

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

Imad Zeroual, Dirk Goldhahn, Thomas Eckart, and Abdelhak Lakhouaja. 2019. OSIAN: Open Source International Arabic News Corpus - Preparation and Integration into the CLARIN-infrastructure. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 175–182, Florence, Italy. Association for Computational Linguistics.

Xinyu Zhang, Xueguang Ma, Peng Shi, and Jimmy Lin. 2021. Mr. TyDi: A multi-lingual benchmark for dense retrieval. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 127–137, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xinyu Zhang, Nandan Thakur, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Mehdi Rezagholizadeh, and Jimmy Lin. 2022a. Making a miracl: Multilingual information retrieval across a continuum of languages.

Xinyu Zhang, Nandan Thakur, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Mehdi Rezagholizadeh, and Jimmy Lin. 2022b. Making a miracl: Multilingual information retrieval across a continuum of languages. *Preprint*, arXiv:2210.09984.

A ArMistral Training

ArMistral, is an autoregressive pretrained language model based on Mistral-7B.

Pretraining data We further pretrain it on a large and diverse Arabic dataset, including all categories of Arabic, namely Classical Arabic (CA), Dialectal Arabic (DA), and MSA. This data is aggregated from various sources: AraNews$_{\text{v2}}$ (Nagoudi et al., 2020), El-Khair (El-Khair, 2016), Gigaword,[2] OSCAR (Suárez et al., 2019), OSIAN (Zeroual et al., 2019), 101 Billion arabic words (Aloui et al., 2024), Wikipedia Arabic, and Hindawi Books.[3] We also derived ArabicWeb22 (A) and (B) from the open source Arabic text 2022.[4] This pretraining dataset was cleaned, filtered and deduplicated using Bhatia (2023). We have also ensured that the model is pretrained in multiple domains, enhancing its results as seen in Table 8.

Instruction Finetuning. To enhance the capabilities of our ArMistral, we instruct-tuning it on three datasets: Alpaca-GPT4, Evol-instruct, and ShareGPT extracted from MultilingualSIFT datasets (Chen et al., 2023).

[2]LDC Catalog Link
[3]OpenITI corpus (v1.6) (?).
[4]ArabicText-2022 data

Alignment Dataset We collected an alignment dataset from Quora and Mawdoo websites and then we took the gold answers as the choosen and we generated the rejected using AceGPT-7B (Huang et al., 2024).

Results

As seen from Table 8, Our ArMistral-Chat model outperforms all existing Arabic LLMs.

B Training methodology

Given a relevant query-document pair (q^+, d^+), we modify the query by appending an instructional template to it. This process transforms the original query q^+ into a new form q^+_{inst} as defined below:

$$q^+_{\text{inst}} = \text{Instruction: }\{\text{task\_instruction}\}\text{ Query:}\{q^+\}$$

Here, "*{task_instruction}*" refers to a one-sentence description of the embedding task taken from Table 10, which outlines the instructions for different tasks. Using a pretrained large language model (LLM), we append a [EOS] token at the end of both the modified query and the document. These are then input into the LLM to extract embeddings $\mathbf{h}_{q^+_{\text{inst}}}$ and $\mathbf{h}_{d^+}$ from the vector at the last [EOS] layer. The training of the embedding model is conducted using the InfoNCE loss function (van den Oord et al., 2019), which is widely recognized for its effectiveness in learning high-quality embeddings. The objective is minimized using the following formulation:

$$\min \left(-\log \frac{\phi(q^+_{\text{inst}}, d^+)}{\phi(q^+_{\text{inst}}, d^+) + \sum_{n_i \in \mathbb{N}} \phi(q^+_{\text{inst}}, n_i)} \right)$$

In the equation above, $\mathbb{N}$ denotes the set of negative samples, and $\phi(q, d)$ is the similarity scoring function between a query q and a document d.

C Datasets overview

The table 9 provides a comprehensive summary of the various datasets utilized in the study. It categorizes datasets based on their type, such as Reranking, Bitext Mining, Retrieval, Crosslingual Retrieval, STS, Pair Classification, Clustering, and Classification. Each entry specifies the dataset name, language, citation, and category, reflecting the diversity and scope of data sources for evaluating the model's performance across different tasks and linguistic contexts.

Model	ARC	Hellaswag	Exams	MMLU	Truthfulqa	ACVA	AlGhafa	Average
ArMistral-7B-Chat	<u>43.20</u>	<u>55.53</u>	<u>45.54</u>	**43.50**	**52.44**	<u>77.06</u>	**35.57**	**50.41**
Jais-13b-chat	41.10	**57.70**	**46.74**	<u>42.80</u>	47.48	72.56	<u>34.42</u>	<u>48.97</u>
AceGPT-13B-chat	**43.80**	52.70	42.09	41.10	<u>49.96</u>	**78.42**	31.95	48.57
AceGPT-13B-base	39.90	51.30	39.48	40.50	46.73	75.29	30.37	46.22
AraLLama-7B-Chat	39.45	50.23	38.24	41.03	50.44	70.45	32.54	46.05
ArMistral-7B-Base	41.50	52.50	38.92	37.50	51.27	69.64	30.24	45.94
Jais-13b-base	39.60	50.30	39.29	36.90	50.59	68.09	30.07	44.98
AceGPT-7B-chat	38.50	49.80	37.62	34.30	49.85	71.81	31.83	44.81
AraLLama-7B-Base	38.40	50.12	38.43	40.23	45.32	69.42	31.52	44.78
AceGPT-7B-base	37.50	48.90	35.75	29.70	43.04	68.96	33.11	42.42

Table 8: Comparison of ArMistral with other Arabic LLMs

D Polydedupe: versatile cleaning Pipeline

PolyDeDupe is a Python package designed for efficient and effective data deduplication across over 100 languages. It supports syntactic and semantic deduplication, making it a versatile tool for high-quality data preprocessing in NLP tasks. Key features include customizable Jaccard similarity thresholds, a performance speed twice that of other tools like SlimPajama, and support for deduplicating instruction tuning data. It can be easily installed via pip to deduplicate datasets, display original and filtered dataset sizes, and identify duplicate clusters. Supported languages span Western, Central, and Eastern European languages, Slavic languages using Cyrillic script, Greek, various Arabic and Devanagari script languages, and more.

E Prompts for evaluation

Table 10 provides an overview of the prompts used for evaluating various tasks. It includes instructions for Reranking, Bitext Mining, Retrieval, Crosslingual Retrieval, Semantic Textual Similarity (STS), Pair Classification, Clustering, and Classification. Each entry outlines the specific task and the corresponding instruction used to guide the model's evaluation process.

F Full Leaderboard

Table 11 presents the performance comparison of various models on different tasks within the ArMTEB benchmark. It includes metrics for Retrieval, Semantic Textual Similarity (STS), Pair Classification (PairCLF), Classification (CLF), Reranking, Clustering, and Bitext Mining (BTM). The table lists each model, its dimensionality, and the scores for each task, along with an overall average score. The results highlight the strengths and weaknesses of each model across a range of tasks, providing a comprehensive overview of their performance.

Type	Dataset	Language	Citation	Category
Reranking	Miracl	Multilingual (Arabic subset)	Zhang et al. (2022b)	s2p
	Mmarco Dev set	Arabic	Bonifacio et al. (2021b)	s2p
	MedicalQA	Arabic	Our Paper	s2p
	MMarco Crosslingual	English to MSA	Bonifacio et al. (2021b)	s2p
	MMarco Crosslingual	MSA to English		s2p
BitextMining	Machine Translation	Moroccan Dialect to English	Nagoudi et al. (2023)	s2s
		Arabizi to French		s2s
		English to MSA		s2s
		French to MSA		s2s
		Spanish to MSA		s2s
		Russian to MSA		s2s
	Code Switching	Algerian Dialect to French	Nagoudi et al. (2023)	s2s
		Egyptian Dialect to English		s2s
		Jordanian Arabic to English		s2s
		Moroccan Arabic to French		s2s
		Palestinian Arabic to English		s2s
		Yemeni Arabic to English		s2s
Retrieval	MLDR	Multilingual (Arabic subset)	Nagoudi et al. (2023)	s2p
	XPDA	Multilingual (Arabic subset)		s2s
	Mintaka	Multilingual (Arabic subset)		s2s
	LareqaQA	Arabic		s2p
	DawqsQA	Arabic		s2s
	ExamsQA	Arabic		s2s
	MKQA	Arabic		s2s
	MLQA	Arabic		s2s
	ARCDQA	Arabic		s2s
	TyDiQA	Arabic		s2s
	XSquadQA	Arabic		s2s
Crosslingual Retrieval	Mmarco Dev set	MSA to German	Bonifacio et al. (2021b)	s2p
		MSA to English		s2p
		MSA to Spanish		s2p
		MSA to Hindi		s2p
		MSA to Vietnamese		s2p
		MSA to Chinese		s2p
		German to MSA		s2p
		English to MSA		s2p
		Spanish to MSA		s2p
		Hindi to MSA		s2p
		Vietnamese to MSA		s2p
		Chinese to MSA		s2p
STS	STS17	Arabic		s2s
	STS22	Arabic		p2p
	Arabic STS Sentence	Arabic		s2s
	Arabic STS Mutli Dialect	Arabic	Our Paper	s2s
	Arabic STS Paragraphs	Arabic		p2p
PairClassification	Xnli	Arabic	Conneau et al. (2018)	s2s
	Orca STS	Arabic	Cer et al. (2017a)	s2s
	M2Q2	Arabic	Elmadany et al. (2022)	s2s
Clustering	Arabic News Paragraphs	Arabic	Our Paper	p2p
	Arabic News headlines	Arabic		s2s
	Baly Stance Paragraphs	Arabic	Baly et al. (2018b)	p2p
	Baly Stance Headings	Arabic	Baly et al. (2018b)	s2s
Classification	Massive Intent	Multilingual (Arabic subset)	FitzGerald et al. (2022)	s2s
	Massive Scenario	Multilingual (Arabic subset)	FitzGerald et al. (2022)	s2s
	Sentiment Analysis	Arabic	Elmadany et al. (2022)	s2s
	Dialect Region	Arabic		s2s
	Dialect Binary	Arabic		s2s
	Dialect Country	Arabic		s2s
	ANS Claim	Arabic		s2s
	Machine Generation	Arabic		s2s
	Age	Arabic		s2s
	Gender	Arabic		s2s
	Adult	Arabic		s2s
	Dangerous	Arabic		s2s
	Emotion	Arabic		s2s
	Hate Speech	Arabic		s2s
	Offensive	Arabic		s2s
	Irony	Arabic		s2s
	Sarcasm	Arabic		s2s
	Abusive	Arabic		s2s

Table 9: Datasets Overview.

Task	Instructions
Reranking	Given an Arabic search query, retrieve web passages that answer the question in {Lang}. Query:{query}.
BitextMining	Retrieve parallel sentences in {Lang}.
Retrieval	Given an Arabic search query, retrieve web passages that answer the question. Query:{query}.
Crosslingual Retrieval	Given an Arabic search query, retrieve web passages that answer the question in {Lang}. Query:{query}.
STS	Retrieve semantically similar text. Text: {text}.
Pair Classification	Retrieve texts that are semantically similar to the given text. Text: {text}.
Clustering	Identify the topic or theme of the given news article. Article:{article}.
Classification	Classify the text into the given categories {options}.

Table 10: Prompts used for evaluation.

Model	Dim.	Retrieval	STS	PairCLF	CLF	Re-rank	Cluster	BTM	Avg
Number of datasets		**23**	**5**	**3**	**18**	**5**	**4**	**12**	**70**
Swan-Large	4096	**65.63**	<u>59.10</u>	**75.62**	52.55	69.42	41.24	**71.24**	**62.11**
multilingual-e5-large	1024	<u>64.01</u>	**59.45**	<u>75.06</u>	53.43	**70.79**	<u>42.49</u>	66.33	<u>61.65</u>
e5-mistral-7b-instruct	4096	56.34	57.02	70.24	53.21	66.24	39.44	<u>70.50</u>	59.00
Swan-Base	768	58.42	58.44	74.93	**57.34**	68.43	40.43	42.45	57.21
multilingual-e5-base	768	56.91	57.99	74.30	52.30	<u>69.07</u>	**42.56**	33.90	55.29
multilingual-e5-small	384	55.14	56.73	73.97	50.85	67.92	42.37	38.47	55.06
LaBSE	768	34.98	54.15	70.60	49.57	62.17	41.42	33.28	49.45
text2vec-base	384	27.69	59.37	71.41	47.94	57.76	37.26	38.32	48.54
ARBERTv2	768	15.12	37.88	62.87	<u>56.85</u>	62.21	39.25	1.99	39.45
CamelBERT-msa	768	9.21	47.69	67.43	55.77	60.20	39.89	1.85	40.29
arabertv02-large	1024	7.34	34.26	63.63	54.32	56.71	37.26	10.97	37.78
arabertv02-base	768	8.62	39.77	66.30	55.77	60.03	41.74	0.70	38.99
CamelBERT-mix	768	7.19	46.47	67.23	56.68	57.50	38.72	0.41	39.17
MARBERTv2	768	5.88	45.21	70.89	54.89	58.64	40.81	0.45	39.54
ARBERT	768	8.07	29.89	61.86	56.92	61.09	37.10	2.28	36.74
CamelBERT-da	768	4.07	41.05	65.82	53.75	54.44	37.63	0.31	36.72
MARBERT	768	2.22	40.62	66.46	54.35	53.09	36.33	0.40	36.21
CamelBERT-ca	768	2.74	36.49	62.26	46.26	51.34	35.77	0.09	33.56

Table 11: ArMTEB Results.