Evaluating Prompt Relevance of Arabic Essays: Insights from Synthetic and Real-World Data

Chatrine Qwaider,[1] Kirill Chirkunov,[1] Bashar Alhafni,[1]
Nizar Habash,[1,2] Ted Briscoe[1]
[1]MBZUAI, [2]New York University Abu Dhabi
{chatrine.qwaider,kirill.chirkunov,bashar.alhafni,ted.briscoe}@mbzuai.ac.ae
nizar.habash@nyu.edu

Abstract

Prompt relevance is a critical yet underexplored dimension in Arabic Automated Essay Scoring (AES). We present the first systematic study of binary prompt-essay relevance classification, supporting both AES scoring and dataset annotation. To address data scarcity, we built a synthetic dataset of on-topic and off-topic pairs and evaluated multiple models, including threshold-based classifiers, SVMs, causal LLMs, and a fine-tuned masked SBERT model. For real-data evaluation, we combined QAES with ZAEBUC, creating off-topic pairs via mismatched prompts. We also tested prompt expansion strategies using AraVec, CAMeL, and GPT-4o. Our fine-tuned SBERT achieved 98% F1 on synthetic data and strong results on QAES+ZAEBUC, outperforming SVMs and threshold-based baselines and offering a resource-efficient alternative to LLMs. This work establishes the first benchmark for Arabic prompt relevance and provides practical strategies for low-resource AES.

1 Introduction

Prompt relevance, or the degree to which an essay responds to its prompt, remains a critical yet understudied factor in Automated Essay Scoring (AES), particularly for Arabic. It captures a learner's task alignment and comprehension, while also supporting trait-specific scoring and filtering off-topic essays for annotation (Persing and Ng, 2014; Cummins et al., 2016). Despite its value, prompt relevance has received limited attention, particularly for Arabic. English-language studies typically handle it implicitly, using feature-based (Persing and Ng, 2014), sentence-level (Rei and Cummins, 2016), or embedding-based approaches (Albatarni et al., 2024). Arabic, however, faces additional challenges like short prompts, topic drift, and a lack of annotated data. Existing Arabic AES work mainly targets holistic scoring (Lotfy et al., 2023; Ghazawi and Simpson, 2025), with no efforts explicitly modeling relevance.

Our goal is to build and evaluate models for prompt relevance classification. We focus on detecting whether a student's essay addresses a given prompt, using a combination of manual annotations, prompt expansion techniques, and relevance classification models.

During dataset construction, a relevance classifier can serve as a prefilter to automatically detect and exclude off-topic essays before annotation. This reduces annotation cost and effort, minimizes noise, and ensures that both trait-specific and holistic scoring models are trained only on essays aligned with their prompts. This is especially important consideration in low-resource contexts like Arabic AES, where manual annotation is costly.

Within AES systems, the relevance classifier can operate as a first-stage module, passing only relevant essays to the scoring module. This prevents inflated or misleading scores for off-topic essays, thereby enhancing the validity and reliability of educational assessments.

To the best of our knowledge, this is the first study to explicitly model prompt relevance in Arabic. Our contributions are as follows:

- We construct prompt-relevance annotations for previously unannotated Arabic datasets to enable supervised modeling.
- We compare several prompt expansion techniques to enhance essay-prompt alignment.
- We propose and evaluate multiple classification approaches, including threshold-based, SVM, causal LLMs, and a fine-tuned masked transformer-based model for prompt-essay relevance classification.

The paper is organized as follows: §2 reviews related work; §3 describes the datasets; §4 outlines prompt expansion strategies; §5 presents our classification methods; and §6 reports results.

Proceedings of The Third Arabic Natural Language Processing Conference, pages 162–178
November 8-9, 2025

2 Related Works

Prompt relevance has received limited attention in the AES literature, despite its importance for both trait-specific scoring and data quality control. Early work in English AES modeled this aspect using feature-based methods. Persing and Ng (2014) introduced prompt adherence modeling with SVMs using lexical and semantic features, while Mathias and Bhattacharyya (2018) used random forests to assess holistic and trait-level score.

Early methods also explored prompt-essay similarity using traditional retrieval techniques. Cummins et al. (2016) computed cosine similarity between TF-IDF vectors of essays and expanded prompts, where expansion terms were generated via random indexing, CBoW, and pseudo-relevance feedback. More recently, Albatarni et al. (2024) proposed a dense retrieval approach using Contriever embeddings to model essay–prompt similarity without feature engineering, achieving state-of-the-art results. This highlights the potential of embedding-based methods for semantic alignment.

In Arabic AES, QAES (Bashendy et al., 2024) is the only publicly available dataset annotated for multiple traits, including prompt relevance. Recent systems such as Lotfy et al. (2023) and Ghazawi and Simpson (2025) focus solely on holistic scoring using BERT-based models, without trait-specific annotations.

To improve cross-prompt robustness, recent models integrate prompt information during training (Li and Ng, 2024), and adopt contrastive and meta-learning techniques to generalize across prompt distributions in low-resource settings (Chen and Li, 2024). Although not always termed prompt expansion, these approaches improve prompt representations to better model topical relevance and the alignment of the essays.

3 Datasets

QAES The QAES dataset (Bashendy et al., 2024), built on the Qatari Corpus of Argumentative Writing (QCAW) (Ahmed et al., 2024), is the only publicly available Arabic AES resource with trait-specific annotations, including prompt relevance[1]. However, it contains only two semantically similar prompts (Telecommunication and Technology), with a skewed distribution favoring relevant essays. We excluded the ambiguous "partially relevant" PR class from our experiments, due to many reasons such that the PR label is inconsistently applied and often ambiguous, and we found essays addressing multiple prompts labeled as PR, further complicating interpretation.Table 1 shows the label distribution.

	R	NR	PR	Total
Train	39	1	18	58
Dev	24	0	15	39
Test	63	3	32	98
Total	**126**	**4**	65	195

Table 1: QAES dataset statistics. **R** (Relevance), **NR** (Non-relevance), **PR** (Partial Relevance).

CEFR Level	Count	Percentage
A2	7	3%
B1	110	51%
B2	80	37%
C1	11	5%
Unassessable	6	3%
Total	214	100%

Table 2: ZAEBUC corpus CEFR level distributions.

ZAEBUC ZAEBUC (Habash and Palfreyman, 2022) is a bilingual Arabic-English dataset of 214 essays written by first-year university students at Zayed University, UAE[2]. Covering three diverse prompts (Social Media, Tolerance, Development), it offers broader topical coverage than QAES. Essays are manually annotated with CEFR levels but lack explicit prompt-essay relevance labels. Table 2 shows the CEFR distribution.

Essay Filtering To verify prompt–essay alignment, we used a GPT-based classifier to predict the most likely prompt for each essay and we compared it to the original assignment. Essays referencing multiple prompts were excluded to ensure a clean relevance signal, yielding a final set of 176 essays. For each, we generated off-topic examples by duplicating the essay and randomly reassigning a different prompt, labeling the pair as non-relevant.

Merged Set (QAES + ZAEBUC) To overcome the limited prompt diversity in QAES and enhance model generalization, we merged QAES with the filtered ZAEBUC dataset. The resulting combined dataset includes five distinct prompts, providing broader coverage of topics and essay styles.

[1] https://gitlab.com/bigirqu/qaes

[2] http://www.zaebuc.org/

	R	NR	Total
Train	2280	2280	4560
Dev	480	480	960
Test	460	460	920

Table 3: Synthetic dataset. **R** (Relevance), **NR** (Non-relevance).

	R	NR	Total
QAES	126	130	256
ZAEBUC	176	176	352
QAES + ZAEBUC	302	306	608
Sythetic data	3220	3220	6440

Table 4: Relevance dataset statistics. **R** (Relevance), **NR** (Non-relevance).

3.1 Synthetic Dataset

We use a synthetic dataset[3] of 3,220 GPT-4o-generated essays in response to 155 prompts across CEFR levels (Qwaider et al., 2025). To simulate large-scale relevance classification, each essay was duplicated, with one paired with its original prompt (relevant) and the other with a randomly selected prompt (non-relevant).

The synthetic dataset was split at the prompt level, with each split (train/dev/test) containing a unique set of prompts and essays. There is no prompts/essays overlap between splits, and each was processed independently when creating the on/off-topic relevance pairs to ensure no cross-split contamination. Table 3 presents the distribution of the two relevance classes across the train, development, and test splits.

Due to the scarcity of large-scale annotated data, this synthetic train-set serves as the main training resource. The development set is used only for hyperparameter tuning, and early stopping. We evaluate models on the QAES dataset, the combined QAES+ZAEBUC dataset, and the synthetic test set to assess generalisation across real and synthetic data (see Table 4).

4 Prompt Expansion Methods

Short prompts often lack semantic depth, reducing the effectiveness of similarity-based methods (Cummins et al., 2016). To enhance their meaning, we apply five expansion strategies, clustering each prompt with semantically related terms.The original prompts range in length from 3 to 26 words; therefore, we apply expansions to the all prompts to ensure experimental consistency. The unexpanded prompt is used as a baseline.

AraVec We applied word-level expansion using the AraVec Wikipedia-SkipGram model (Soliman et al., 2017). Each prompt was first tokenized, and cleaned by removing stopwords. For each remaining word, we retrieved its top five most similar words based on cosine similarity in the AraVec embedding space. Out-of-vocabulary (OOV) words were marked accordingly.

CAMeLBERT We applied a contextualized prompt expansion using CAMeLBERT (Inoue et al., 2021). Prompts were tokenized using the CAMeL tokenizer, and each word was masked in context to generate the top five substitutes via a fill-mask pipeline.

POS-Aware Prompt Expansion We implemented two POS-aware prompt expansion strategies using Arabic linguistic tools. In the ARAVEC_POS method, we use the CAMeLBERT Disambiguator (Inoue et al., 2022) for part-of-speech tagging and retrieve the top 10 similar words from AraVec for nouns and the top 5 for other POS tags, prioritizing content-rich terms. The CAMeLBERT_POS method follows the same POS-guided approach but uses CAMeLBERT as a masked language model, combining contextual predictions with linguistic relevance to produce richer, POS-sensitive expansions.

GPT-4o Expansion We used GPT-4o for structured prompt expansion by generating five subheaders per Arabic prompt. For every subheader, the model was instructed to suggest five relevant clue words that students might use. This approach provides topic-focused, semantically rich prompt expansions. The prompt used for this task is shown in Figure 2 (Appendix C), and a full example is provided in Appendix A.

5 Methodology

5.1 Semantic Similarity Modeling

To model prompt–essay semantic relationships, we use sentence embeddings from various pretrained language models. For each model, we extract vector representations for both the **essay** and its corresponding (original or expanded) **prompt**. The language models employed include Arabic-specific models such as CAMeLBERT (Inoue et al.,

[3] https://github.com/mbzuai-nlp/arabic-aes-bea25

Model	Version / Source	Size
CaMELBERT	bert-base-arabic-camelbert-mix	110M
AraBERT	AraBERTv0.2-base	136M
SBERT	paraphrase-multilingual-MiniLM-L12-v2	118M
MARBERT	UBC-NLP/MARBERT	163M
ARBERT	UBC-NLP/ARBERT	163M
Matryoshka STS	omarelshehy/arabic-english-sts-matryoshka-v2.0	560M
MoE	nomic-ai/nomic-embed-text-v2-moe	305M
LaBSE	LaBSE	471M
DistilBERT-based	distiluse-base-multilingual-cased-v1	135M
Multilingual BERT	bert-base-multilingual-cased	179M

Table 5: Embedding models used in our experiments along with their sizes.

Model	Expansion Method	Avg R	Avg NR	Diff	Stdev R	Stdev NR
PMMLM12v2	**Original**	0.7100	0.3065	**0.4035**	0.1337	0.1246
	Aravec	0.6503	0.3060	0.3443	0.1310	0.1247
	CAMEL	0.5666	0.2736	0.2930	0.1387	0.1174
	Aravec_POS	0.6407	0.3105	0.3302	0.1467	0.1288
	CAMEL_POS	0.6159	0.2948	0.3211	0.1412	0.1172
	GPT	0.6999	0.3228	**0.3771**	0.1324	0.1520
NETv2-m	**Original**	0.6479	0.2925	**0.3554**	0.0867	0.0606
	Aravec	0.6037	0.3464	0.2574	0.0815	0.0544
	CAMEL	0.6019	0.3622	0.2397	0.0832	0.0581
	Aravec_POS	0.6138	0.3621	0.2517	0.0833	0.0548
	CAMEL_POS	0.6358	0.3906	0.2451	0.0836	0.0614
	GPT	0.7455	0.4180	**0.3275**	0.1016	0.0787
DBMCv1	**Original**	0.4381	0.1174	**0.3207**	0.1019	0.1008
	Aravec	0.4805	0.1860	0.2945	0.1037	0.1009
	CAMEL	0.4823	0.1900	0.2923	0.0910	0.1030
	Aravec_POS	0.4797	0.1958	0.2839	0.0953	0.1018
	CAMEL_POS	0.4760	0.2015	0.2745	0.0913	0.1033
	GPT	0.5542	0.2081	**0.3461**	0.0957	0.1192

Table 6: Cosine similarity statistics across models and prompt expansion methods in the synthetic test-set. **R** (Relevance), **NR** (Non-relevance), **Diff** (Difference),**PMMLM12v2** (paraphrase-multilingual-MiniLM-L12-v2), **NETv2-m** (nomic-embed-text-v2-moe), **DBMCv1** (distiluse-base-multilingual-cased-v1).

2021), AraBERT (Antoun et al.), MARBERT, and ARBERT (Abdul-Mageed et al., 2021); multilingual models like mBERT (Devlin et al., 2018), LaBSE (Feng et al., 2022), and DistilUSE (Yang et al., 2019); as well as cross-lingual and Semantic Textual Similarity STS optimised models including SBERT (Reimers and Gurevych, 2019), Matryoshka (Kusupati et al., 2024), and the Mixture of Experts model (Nussbaum and Duderstadt, 2025). Table 5 shows the used LMs.

We start by evaluating on the synthetic dataset. For each LM and expansion method, we compute cosine similarity between prompt and essay embeddings. We report the mean and standard deviation of semantic similarity per class (ON/OFF), using the mean difference as a discriminative indicator. Table 6 highlights the top results while the full results are in Appendix B. Among all expansion methods, the original prompt and GPT-based expansion consistently achieved the highest separation between relevance classes across models. Based on these results, we retain these two settings for subsequent experiments. Tables 7 and 8 present the results for the top models in the two most effective prompt settings evaluated in the QAES dataset and the combined QAES+ZAEBUC dataset, respectively. Among all evaluated models, the (paraphrase-multilingual-MiniLM-L12-v2) model achieved the highest class separation across both prompt settings. Based on these results, we

Model	Expansion	Avg R	Avg NR	Diff	Stdev R	Stdev NR
PMMLM12v2	**Original**	0.6322	0.3356	**0.2967**	0.1211	0.0919
	GPT	0.5849	0.4637	0.1213	0.1178	0.0993
NETv2-m	Original	0.6402	0.3803	0.2599	0.0486	0.0488
	GPT	0.5982	0.4613	0.137	0.0860	0.0731
DBMCv1	Original	0.3778	0.1006	0.2772	0.1325	0.0868
	GPT	0.3484	0.2615	0.0869	0.0991	0.0891

Table 7: Cosine similarity statistics across models and prompt expansion methods in the QAES dataset. **R** (Relevance), **NR** (Non-relevance), **PMMLM12v2** (paraphrase-multilingual-MiniLM-L12-v2), **NETv2-m** (nomic-embed-text-v2-moe), **DBMCv1** (distiluse-base-multilingual-cased-v1).

Model	Expansion Method	Avg R	Avg NR	Diff	Stdev R	Stdev NR
PMMLM12v2	**Original**	0.6919	0.325	**0.3669**	0.1515	0.1077
	GPT	0.6014	0.3882	0.2132	0.1394	0.1241
NETv2-m	Original	0.6278	0.3029	0.3249	0.0604	0.0899
	GPT	0.5679	0.3841	0.1838	0.0838	0.1031
DBMCv1	Original	0.4131	0.1323	0.2809	0.1244	0.1009
	GPT	0.3559	0.1893	0.1667	0.1026	0.1177

Table 8: Cosine similarity statistics across models and prompt expansion methods in the QAES+ZAEBUC dataset. **R** (Relevance), **NR** (Non-relevance), **PMMLM12v2** (paraphrase-multilingual-MiniLM-L12-v2), **NETv2-m** (nomic-embed-text-v2-moe), **DBMCv1** (distiluse-base-multilingual-cased-v1)

retain the paraphrase-multilingual-MiniLM-L12-v2 model for subsequent experiments, with further analysis provided in §6. These measurements provide insight into the semantic separability of relevant and non-relevant pairs and serve as a foundation for threshold-based and classification models.

5.2 SVM Classifier

To establish a baseline beyond cosine similarity, we train a Support Vector Machine (SVM) classifier using the synthetic dataset. This setup enables us to evaluate the effectiveness of discriminative modelling compared to raw embedding similarity. Each input to the model consists of the concatenated embeddings of the essay and its corresponding prompt. In an alternative setting, the cosine similarity between these embeddings is also included as an additional feature. The SVM is trained on the synthetic training data and evaluated across three datasets.

5.3 Threshold Classifier

As a simpler alternative to supervised learning, we implement a threshold-based classifier using cosine similarity between prompt and essay embeddings. To set the threshold we compute the mean cosine similarity for relevant pairs (avg_sim) and non-relevant pairs (avg_dis) on the development split. As a lightweight baseline, the decision threshold is set to the midpoint between these two means, providing a transparent and reproducible reference point. This fixed threshold is then applied to the held-out test set for evaluation[4]. The classifier operates under a simple decision rule: if the similarity score exceeds the threshold, it predicts relevant; otherwise, it predicts not relevant. This approach provides a reference point for comparing the effectiveness of embedding-based similarity against more complex classifiers such as SVMs and LMs.

5.4 LLMs as classifiers

To explore how far the latest generation of small causal LLMs (<7B parameters) can meet this need in Arabic, we adapt a range of recently released open-weight checkpoints as essay-prompt relevance classifiers through prompt-engineering strategies and map free-form responses to relevant/not-relevant labels. This setup allows us to directly compare how these small LLMs perform against embedding-based methods, SVMs, and fine-tuned masked transformer models on the same task. Small LLMs set consists of ten open-weight model

[4]For example, paraphrase-multilingual-MiniLM-L12-v2, relevance mean = 0.7, non-relevance = 0.3, making 0.5 a reasonable decision boundary.

versions between 0.5B and 6.7B parameters with Arabic support published in the last year. Gemma 3 series includes the 1B and 4B instruction-tuned modern decoder-only models. The Falcon H1 (hybrid architecture: Attention + SSM, Mamba 2) (Falcon-LLM-Team, 2025) contributes a 0.5B instruction model and a 1.5B version with a reasoning feature, allowing us to test whether extra steps improve relevance judgments. Qwen 3 (An Yang, 2025) adds 0.6B and 1.7B checkpoints, both exploited with "thinking mode" chain-of-thought support. Finally, the Arabic-centric Jais-Family (Sengupta et al., 2023) (Inception, 2024) offers a smooth size ladder - 560 M, 1.3B, 2.7B, and 6.7B chat models.

We treat topic relevance as a binary question-answering task framed through chat completion. For each essay, the model is prompted with a task definition, an (expanded) prompt, and the essay text, ending with: "*Is the essay relevant to this topic? Answer Yes or No.*" The prompt includes two-shot examples (one relevant and one not). We use models as-is, without fine-tuning, and convert their free-form responses into binary labels: "Yes-like" ⇒ 1 (relevant), "No-like" ⇒ 0. An English-translated prompt schema is in Appendix E. For models with built-in reasoning modes (e.g., Qwen's "thinking" mode, Falcon-H1's reasoning variant), we enable them to support multi-step logic. All models use conservative decoding settings: low temperature (0.3), high top-p (0.8), and generation restricted to a single token.Despite this, responses vary ranging from English ("yes", "no") to transliterated Arabic ("na'am", "laa") or numeric forms (1, 0, -1). We map outputs to binary labels: affirmative forms map to 1 (relevant), and negative forms to 0 (not relevant). Unrecognized responses map to 0.

5.5 Fine-Tuned Language Models

To enhance relevance modeling beyond static embeddings, we fine-tune a SBERT model using our synthetic dataset. The goal is to learn more expressive semantic representations that capture the alignment between prompts and essays. We use a cosine similarity loss to directly optimize the model's embedding space such that semantically related prompt-essay pairs are brought closer together. We conduct experiments on both the original and GPT-expanded prompts using the best-performing paraphrase-multilingual-MiniLM-L12-v2 model.

Evaluated the model across three test conditions and two prompt configurations. These evaluations allow us to assess the model's ability to generalize beyond the synthetic domain and determine whether supervised fine-tuning improves relevance detection over the baseline SVM model and compared to a simple threshold approach.

Table 15, in Appendix G summarizes the hyperparameter settings used across all models.

Model / Version	Size
FalconH1	
Falcon-H1-0.5B-Instruct	997M
Falcon-H1-1.5B-Deep-Instruct	**3.0G**
Qwen3	
Qwen3-0.6B	1.5G
Qwen3-1.7B	3.8G
Gemma-3	
gemma-3-1b-it	1.9G
gemma-3-4b-it	8.1G
Jais-Family	
jais-family-590m-chat	2.9G
jais-family-1p3b-chat	5.9G
jais-family-2p7b-chat	12G
jais-family-6p7b-chat	27G

Table 9: Small LLMs used in our experiments along with their sizes.

6 Results and Discussion

6.1 Semantic Similarity Modeling

We evaluated cosine similarity scores to compare models and expansion strategies. Original and GPT-4o-expanded prompts showed the best class separation, for instance, SBERT achieved gaps of 0.4035 (original) and 0.3771 (GPT), outperforming Aravec (0.3443) and CAMeLBERT (0.2930), (see Tables 6,12). These results expose the limitations of non-contextual embeddings like Aravec, which often retrieve off-topic words due to OOV issues and lack of contextual awareness, especially in short prompts (Mikolov et al., 2013). CAMeLBERT, while leveraging masked language modeling, can fail in short-text contexts. For example, when key tokens هواية (hobby) is masked in " تحدث عن هواية تحبها " (Talk about a hobby you like), the model can retrieve generic or unrelated terms like رياضة (sport) or دولة (country), which may not fit well in context. Such substitution noise reduces semantic precision. GPT-4o-based expansions outperform other strategies, likely due to their

Dataset (Prompts)	Syn				Q				QZ			
	Original		GPT		Original		GPT		Original		GPT	
Models	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
SVM												
Embedding	73	71	**79**	**77**	59	50	65	46	73	70	57	51
Embedding +SS	88	88	**91**	**91**	51	34	55	44	50	34	52	38
Threshold	**95**	**95**	90	90	89	88	71	74	91	91	82	82
Small LLMs												
FalconH1-1.5B-DI	**97**	**96**	88	90	88	87	80	75	95	94	91	90
Qwen3-1.7B	**97**	**97**	85	83	90	89	76	69	92	92	82	80
Gemma-3-1B-it	**81**	**76**	53	14	61	36	51	0	60	34	51	1
Jais-Family-6p7b	**91**	**90**	80	76	81	77	69	72	81	77	64	64
Fine-Tune SBERT												
PMMLM12v2	**98**	**97**	**98**	**97**	86	85	73	76	91	91	85	86

Table 10: Overall performance comparison across models and methods, including SVM classification, threshold-based classification, small LLMs, and fine-tuned SBERT. Evaluations are conducted on **Syn** (Synthetic test_set), **Q** (QAES), and **QZ** (QAES_ZAEBUC). Reported metrics are Accuracy (Acc) and F1-score (F1) in (%).

semantically rich prompts with subheaders and clue words, which provide stronger contextual grounding for modeling prompt–essay relevance.

In terms of dataset effects, synthetic data shows high class separability (e.g., SBERT = 0.4035), while QAES, limited to two similar prompts, exhibits much smaller gaps (SBERT = 0.2967). Merging with ZAEBUC increases topic diversity and restores separability (SBERT = 0.3669), confirming the benefit of broader prompt coverage. (See Table 7, 8). Finally, Sentence Transformer models like SBERT MiniLM, nomic-MoE, LaBSE outperform others due to their training on STS tasks and use of Siamese architectures tailored for sentence-level comparison, unlike token-focused models as CAMeLBERT or MARBERT. These models also exhibit lower standard deviation, indicating more reliable similarity judgments across domains.

6.2 SVM Classification

To evaluate the effectiveness of traditional supervised models, we built an SVM classifier. Table 10 presents the overall performance of all proposed models. As shown, in the SVM_synthetic setting, adding cosine similarity significantly boosted performance. With GPT-expanded prompts, the F1 score rose from 77% to 91%, while for original prompts, it improved from 71% to 88%. This is expected, given that the synthetic data used for both training and testing shares a consistent structure, generator (GPT-4o), and topical coherence. These conditions make the decision boundary between relevant and non-relevant pairs easier for the model to learn. See Appendix D, Table 13, for the complete evaluation of the synthetic data set across language models.

In real data, however, this advantage does not hold. In the QAES data set, adding cosine similarity reduced F1 performance from 50% to 34% for the original prompts and from 46% to 44% for GPT prompts. On the merged QAES+ZAEBUC dataset, similarity still failed to help, with F1 scores remaining low (38% with GPT + similarity). The best real-world result was achieved using only embeddings with original prompts on the QAES+ZAEBUC dataset (F1 = 70%). This highlights that increasing topic diversity can improve the classifier's ability to learn separable decision boundaries, but only when using the original prompts. In contrast, GPT-expanded prompts introduce additional related words across prompts, which blur the boundaries between relevance classes and confuse the classifier. These results suggest that in supervised models like SVM, prompt expansion can sometimes hurt performance by introducing cross-topic noise, mainly when relevance depends on subtle topic differences. This supports findings that cosine similarity underperforms in dense spaces or with misaligned embeddings (Steck et al., 2024).

6.3 Threshold-Based Classification

We implemented a cosine similarity threshold-based classifier using a fixed threshold of 0.5 applied to sentence embeddings SBERT (paraphrase-multilingual-MiniLM-L12-v2), see Table 10. On the synthetic test set, the cosine similarity thresh-

old classifier achieves strong results, reaching an F1 score of 95% for the original prompt and 90% with the GPT-expanded prompt. These high scores demonstrate the effectiveness of simple similarity-based decisions in controlled, GPT-generated environments where prompt-essay pairs are clearly aligned. In the QAES dataset, performance declines where F1 drops to 88% (original) and 74% (GPT). This decline reflects the compressed semantic margins caused by overlapping prompt topics, where many non-relevant essays still exhibit moderate similarity scores, making them harder to separate using a fixed threshold. Interestingly, performance improves again on the QAES+ZAEBUC dataset, with F1 scores rising to 91% (original) and 82% (GPT). Broader topic diversity improves separability in the embedding space, enhancing thresholding effectiveness. Overall, the threshold-based classifier is lightweight yet competitive—outperforming SVMs on real essays and closely matching advanced models on synthetic data. However, its reliance on a fixed threshold limits its robustness in cases of high semantic overlap or domain shift, underscoring the need for more adaptive approaches.

6.4 Small LLM Classifiers

Table 10 reports the top performing model from each LLM family, while the complete results are provided in Appendix F, Table 14. The results consistently indicate that model size, measured by the number of parameters, is the best indicator of accuracy on topic-essay relevance. On every dataset, the models above the 1B - Falcon-1.5B-DeepInstruct, Qwen3-1.7B, and the larger Jais-Family versions - clearly outperform the SVM baseline. An exception is the Gemma-3 series: despite scaling from 1B to 4B parameters, both versions lag behind the baseline across all three test sets. Adding a GPT-expanded prompt led to a decline in performance for small LLMs. We attribute this to the expansion narrowing the semantic scope: many essays mention the main topic obliquely but omit several of the newly appended keywords, prompting the classifier to over-penalize otherwise relevant answers. Reasoning models like Falcon-H1-1.5B and Qwen3-1.7B, with built-in chain-of-thought capabilities, match or exceed cosine-based classifiers without fine-tuning. They achieve over 96% F1 on synthetic data, 87% on QAES, and 92% on QAES+ZAEBUC, suggesting that self-reasoning aids in identifying core topical cues, even in longer or noisier essays. However, these gains come at a steep computational price: a Falcon-H1-1.5B run needs over 25x the memory footprint (Table 5, 9) and approximately 50x the inference time of SBERT, making it cost-ineffective for large-scale batch processing. Until the current miniaturization trend in LLM research narrows this gap, transformer models still retain the top place in terms of efficiency for ad hoc NLU tasks. At the same time, small LLMs with prompt engineering support could be used for fast prototyping of a solution.

6.5 Fine-Tuned SBERT Model

To move beyond static embeddings and heuristic decision rules, we fine-tuned the SBERT model (paraphrase-multilingual-MiniLM-L12-v2) on the synthetic dataset and evaluated its generalization to real and mixed data settings, results are shown in Table 10. On the synthetic test set, the fine-tuned model achieved an F1 score of 97% for both original and GPT-expanded prompts, reflecting near-perfect alignment modeling. This result is expected, as both the training and test data were generated by GPT-4o and follow similar lexical and topical structures. Moreover, the model was optimized using a cosine similarity loss, which aligns directly with the inference objective. When evaluated on the QAES+ZAEBUC dataset, the fine-tuned SBERT model achieved strong performance, with F1 = 91% using original prompts and 86% with GPT-expanded prompts. This outperforms the SVM baseline and matches or exceeds the performance of the threshold classifier, demonstrating the model's robustness across diverse prompts and writing styles. On the more limited QAES dataset, performance is lower, with F1 = 85% (original) and 76% (GPT). This decline is consistent with previous findings and likely reflects QAES's narrow topical scope and high prompt similarity, which make prompt-essay distinctions harder to learn. Additionally, the small dataset size (195 essays) limits the generalizability and stability of evaluation results. Fine-tuning with cosine similarity loss effectively restructures the embedding space to reflect task-specific alignment, clustering relevant pairs, and pushing apart irrelevant ones, even in cases of lexical overlap. Although this is effective in well-structured or synthetic data, model performance can degrade when exposed to real-world variability. In such cases, domain adaptation or fine-tuning with real annotated data becomes necessary to preserve generalization.

	Acc	F1
Raw Essays	95.7	95.6
Error-Free Essays	96.9	96.8

Table 11: Performance of finetuning the SBERT model on the ZAEBUC dataset, Accuracy (Acc) and F1-score (F1) in (%).

6.6 Generalization Analysis

To check the fine-tunning model robustness, we conducted an experiment on the ZAEBUC. We evaluate the model on the raw student essays containing errors and their crossponding manually corrected versions, under the usage of original prompts. Table 11 presents the results. Evaluation on raw essays shows strong performance (F1-score of 95.6%), while performance on corrected essays is even higher (F1-score of 96.8%).

Although erroneous essays mimic real learner writing, we test whether the model generalizes in an ideal setting. Results show robustness to noisy data and strong performance on corrected essays: when trained on error-injected data, the model also generalizes well to clean text. This suggests it captures underlying linguistic features beyond surface errors.

6.7 Qualitative Differences Between Synthetic and Real-World Data

We also examine qualitative aspects of the data sets to understand the observed performance gap better. Synthetic data exhibits a larger vocabulary size compared to real-world essays (24K vs. 15K), but avoiding rare words and subword tokenization mitigates OOV issues. The fine-tuned model demonstrates robustness to noisy learner input with grammatical errors, suggesting that lexical coverage and surface-level noise are not the primary limiting factors.

However, our analyses on real-world dataset highlight that most accuracy drops are driven by structural shifts rather than vocabulary or noise. Essays in real-world corpora contain longer sentences (median 12 vs 8 words), longer paragraphs (96 vs. 44 words), and fewer paragraph breaks. Misclassifications are concentrated in essays with structural properties far from synthetic medians or containing structural anomalies. These structural mismatches, although affecting only a small subset of samples, explain the residual performance gap between synthetic and real-world evaluations.

7 Conclusion and Future Work

This work presents the first study of prompt-essay relevance modeling for Arabic. We use synthetic data, prompt expansion, and a range of models. Expanded prompts consistently improved the separation of relevant and irrelevant essays, especially in diverse datasets.

Future work will explore graded relevance scoring instead of binary classification, modeling prompt-essay coherence throughout the text, incorporating human annotations, and apply domain-adaptive fine-tuning using real student essays. These extensions will facilitate the effective integration of prompt relevance scores into an Arabic AES system.

Limitations

This study has several limitations. First, the scarcity of manually annotated data constrained model training and evaluation, requiring heavy reliance on synthetic examples. Second, the use of fixed cosine similarity thresholds may not generalize well across different domains or prompt types, potentially limiting their applicability in more diverse contexts. Lastly, the presence of mixed-topic essays and semantically close prompts introduced ambiguity in relevance annotations, which may have affected both training quality and evaluation reliability.

Ethical Considerations

This research employs a combination of publicly available and restricted-access datasets. The synthetic dataset and the ZAEBUC dataset are freely accessible for research use. In contrast, the QAES dataset is not openly available, as it is distributed through the Linguistic Data Consortium under license. All essay texts used were anonymized, with no personally identifiable information included.

References

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

Abdelhamid M Ahmed, Xiao Zhang, Lameya M Rezk, and Wajdi Zaghouani. 2024. Building an annotated l1 arabic/l2 english bilingual writer corpus: the qatari corpus of argumentative writing (qcaw). *Corpus-Based Studies across Humanities*, 1(1):183–215.

Salam Albatarni, Sohaila Eltanbouly, and Tamer Elsayed. 2024. Graded relevance scoring of written essays with dense retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1329–1338.

et al. An Yang. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

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

May Bashendy, Salam Albatarni, Sohaila Eltanbouly, Eman Zahran, Hamdo Elhuseyin, Tamer Elsayed, Walid Massoud, and Houda Bouamor. 2024. Qaes: First publicly-available trait-specific annotations for automated scoring of arabic essays. In *Proceedings of The Second Arabic Natural Language Processing Conference*, pages 337–351.

Yuan Chen and Xia Li. 2024. Plaes: Prompt-generalized and level-aware learning framework for cross-prompt automated essay scoring. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12775–12786.

Ronan Cummins, Helen Yannakoudakis, and Ted Briscoe. 2016. Unsupervised modeling of topical relevance in l2 learner text. In *Proceedings of the 11th workshop on innovative use of NLP for building educational applications*, pages 95–104.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Falcon-LLM-Team. 2025. Falcon-h1: A family of hybrid-head language models redefining efficiency and performance.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Rayed Ghazawi and Edwin Simpson. 2025. How well can llms grade essays in arabic? *Computers and Education: Artificial Intelligence*, 9:100449.

Nizar Habash and David Palfreyman. 2022. Zaebuc: An annotated arabic-english bilingual writer corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 79–88.

Inception. 2024. Jais family model card.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 92–104, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Go Inoue, Salam Khalifa, and Nizar Habash. 2022. Morphosyntactic tagging with pre-trained language models for Arabic and its dialects. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1708–1719, Dublin, Ireland. Association for Computational Linguistics.

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. 2024. Matryoshka representation learning. *Preprint*, arXiv:2205.13147.

Shengjie Li and Vincent Ng. 2024. Conundrums in cross-prompt automated essay scoring: Making sense of the state of the art. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 7661–7681.

Nourmeen Lotfy, Abdulaziz Shehab, Mohammed Elhoseny, and Ahmed Abu-Elfetouh. 2023. An enhanced automatic arabic essay scoring system based on machine learning algorithms. *CMC-COMPUTERS MATERIALS & CONTINUA*, 77(1):1227–1249.

Sandeep Mathias and Pushpak Bhattacharyya. 2018. Asap++: Enriching the asap automated essay grading dataset with essay attribute scores. In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*.

Zach Nussbaum and Brandon Duderstadt. 2025. Training sparse mixture of experts text embedding models. *Preprint*, arXiv:2502.07972.

Isaac Persing and Vincent Ng. 2014. Modeling prompt adherence in student essays. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1534–1543.

Chatrine Qwaider, Bashar Alhafni, Kirill Chirkunov, Nizar Habash, and Ted Briscoe. 2025. Enhancing Arabic automated essay scoring with synthetic data and error injection. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 549–563, Vienna, Austria. Association for Computational Linguistics.

Marek Rei and Ronan Cummins. 2016. Sentence similarity measures for fine-grained estimation of topical relevance in learner essays. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 283–288, San Diego, CA. Association for Computational Linguistics.

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

Neha Sengupta, Sunil Kumar Sahu, Bokang Jia, Satheesh Katipomu, Haonan Li, Fajri Koto, William Marshall, Gurpreet Gosal, Cynthia Liu, Zhiming Chen, Osama Mohammed Afzal, Samta Kamboj, Onkar Pandit, Rahul Pal, Lalit Pradhan, Zain Muhammad Mujahid, Massa Baali, Xudong Han, Sondos Mahmoud Bsharat, and 13 others. 2023. Jais and jais-chat: Arabic-centric foundation and instruction-tuned open generative large language models. *Preprint*, arXiv:2308.16149.

Abu Bakr Soliman, Kareem Eissa, and Samhaa R El-Beltagy. 2017. Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, 117:256–265.

Harald Steck, Chaitanya Ekanadham, and Nathan Kallus. 2024. Is cosine-similarity of embeddings really about similarity? In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, page 887–890. ACM.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, and 1 others. 2019. Multilingual universal sentence encoder for semantic retrieval. *arXiv preprint arXiv:1907.04307*.

A Prompt Expansion

Prompt	تحدث عن أهمية التعليم الرقمي في عصرنا الحالي. .Discuss the importance of digital education in our current era .
Aravec	'تحدث': ['وتحدث', 'يحدث', 'ويحدث', 'حدوث', 'تكون_ناجمه'] 'أهمية': ['<OOV>'], 'التعليم': ['للتعليم', 'والتعليم', 'التعليم_العالي', 'تعليم', 'التعليم_الابتدائي'], 'الرقمي': ['الرقميه', 'رقميه', 'والرقمي', 'البيانات_الرقميه', 'التناظري'], 'عصرنا': ['ايامنا_هذه', 'العصر', 'عصر', 'الحاضر»', 'العصر_الحديث'], 'الحالي': ['السابق', 'الحاليه', 'الجديد', 'جمعيه_ولايه_بجايه', 'الماضي_ليكون_بالشكل'] 'discuss': ['and discussed', 'happens', 'and happens', 'occurrence', 'is caused'], 'importance': ['<OOV>'], 'education': ['for education', 'and education', 'higher education', 'education', 'primary education'], 'digital': ['digital', 'digital', 'and digital', 'digital data', 'analogue'], 'our era': ['these days', 'the era', 'era', 'the present', 'the modern era'], 'current': ['previous', 'current', 'new', 'Association of the State of Béjaïa', 'the past to be in this form']
CaMELBERT	{'تحدث': ['يتحدث', 'ناهيك', 'فضلا', 'الحديث', 'وتحدث'], 'أهمية': ['تقنيات', 'أهمية', 'واقع', 'أثر', 'دور'], 'التعليم': ['الإعلام', 'التحول', 'الاعلام', 'الأمن', 'المحتوى'], 'الرقمي': ['الجامعي', 'الإلكتروني', 'والتدريب', 'العالي', 'المدرسي'], 'عصرنا': ['الوقت', 'العصر', 'العالم', 'عالمنا', 'القرن'], 'الحالي': ['الرقمي', 'الحالي', 'الحديث', 'الحاضر', 'هذا']} 'discuss': ['speaks', 'not to mention', 'moreover', 'talk/discussion', 'and discussed'], 'importance': ['technologies', 'importance', 'reality', 'impact', 'role'], 'education': ['media', 'transformation', 'the media', 'security', 'content'], 'digital': ['university-level', 'electronic', 'and training', 'higher', 'school-based'], 'our era': ['time', 'era', 'world', 'our world', 'century'], 'current': ['digital', 'current', 'modern', 'present', 'this']
GPT	١. تعريف التعليم الرقمي: [التكنولوجيا، الإنترنت، المحتوى، المنصات، الأجهزة]. ٢. فوائد التعليم الرقمي: [التفاعل، المرونة، الوصول، التوفير، التخصيص]. ٣. التحديات التي تواجه التعليم الرقمي: [البنية التحتية، الخصوصية، التكلفة، التدريب، المصداقية]. ٤. تأثير التعليم الرقمي على الطلاب: [الإبداع، التحفيز، الاستقلالية، التعاون، النتائج]. ٥. مستقبل التعليم الرقمي: [الابتكار، التطور، الاتجاهات، الذكاء الاصطناعي، الواقع الافتراضي]. 1. **Definition of Digital Education**: [Technology, Internet, Content, Platforms, Devices]. 2. **Benefits of Digital Education**: [Interactivity, Flexibility, Accessibility, Cost-effectiveness, Personalization]. 3. **Challenges Facing Digital Education**: [Infrastructure, Privacy, Cost, Training, Credibility]. 4. **Impact of Digital Education on Students**: [Creativity, Motivation, Autonomy, Collaboration, Outcomes]. 5. **Future of Digital Education**: [Innovation, Advancement, Trends, Artificial Intelligence, Virtual Reality].

Figure 1: An Example of a prompt with its expansion variations by Aravec, CAMeLBERT, and GPT.

B Semantic cosine similarity modeling

Model	Expansion Method	Avg R	Avg NR	Diff	Stdev R	Stdev NR
paraphrase-multilingual-MiniLM-L12-v2	Original	0.7100	0.3065	0.4035	0.1337	0.1246
	Aravec	0.6503	0.3060	0.3443	0.1310	0.1247
	CAMEL	0.5666	0.2736	0.2930	0.1387	0.1174
	Aravec_POS	0.6407	0.3105	0.3302	0.1467	0.1288
	CAMEL_POS	0.6159	0.2948	0.3211	0.1412	0.1172
	GPT	0.6999	0.3228	0.3771	0.1324	0.1520
nomic-embed-text-v2-moe	Original	0.6479	0.2925	0.3554	0.0867	0.0606
	Aravec	0.6037	0.3464	0.2574	0.0815	0.0544
	CAMEL	0.6019	0.3622	0.2397	0.0832	0.0581
	Aravec_POS	0.6138	0.3621	0.2517	0.0833	0.0548
	CAMEL_POS	0.6358	0.3906	0.2451	0.0836	0.0614
	GPT	0.7455	0.4180	0.3275	0.1016	0.0787
distiluse-base-multilingual-cased-v1	Original	0.4381	0.1174	0.3207	0.1019	0.1008
	Aravec	0.4805	0.1860	0.2945	0.1037	0.1009
	CAMEL	0.4823	0.1900	0.2923	0.0910	0.1030
	Aravec_POS	0.4797	0.1958	0.2839	0.0953	0.1018
	CAMEL_POS	0.4760	0.2015	0.2745	0.0913	0.1033
	GPT	0.5542	0.2081	0.3461	0.0957	0.1192
arabic-english-sts-matryoshka-v2.0	Original	0.6788	0.4056	0.2732	0.1099	0.1258
	Aravec	0.7017	0.4560	0.2458	0.1121	0.1300
	CAMEL	0.6907	0.4572	0.2335	0.1130	0.1372
	Aravec_POS	0.7004	0.4945	0.2059	0.1128	0.1323
	CAMEL_POS	0.6942	0.4761	0.2182	0.1080	0.1392
	GPT	0.7956	0.5180	0.2775	0.1227	0.1515
LaBSE	Original	0.5073	0.3440	0.1633	0.0762	0.0779
	Aravec	0.5923	0.4185	0.1738	0.0736	0.0879
	CAMEL	0.6171	0.4621	0.1549	0.0833	0.0822
	Aravec_POS	0.6150	0.4481	0.1670	0.0757	0.0938
	CAMEL_POS	0.5993	0.4679	0.1313	0.0832	0.0813
	GPT	0.6739	0.4861	0.1879	0.0867	0.0965
ARBERT	Original	0.3642	0.3072	0.0570	0.0416	0.0382
	Aravec	0.5230	0.4615	0.0615	0.0494	0.0474
	CAMEL	0.4772	0.4184	0.0588	0.0432	0.0441
	Aravec_POS	0.5199	0.4682	0.0517	0.0481	0.0455
	CAMEL_POS	0.4675	0.4071	0.0604	0.0464	0.0376
	GPT	0.5521	0.4678	0.0843	0.0497	0.0549
bert-base-arabertv2	Original	0.4898	0.4695	0.0203	0.0805	0.0663
	Aravec	0.7691	0.7127	0.0564	0.0510	0.0466
	CAMEL	0.7554	0.7336	0.0217	0.0435	0.0287
	Aravec_POS	0.7898	0.7472	0.0426	0.0505	0.0480
	CAMEL_POS	0.7725	0.7636	0.0089	0.0506	0.0286
	GPT	0.8333	0.8136	0.0197	0.0392	0.0400
bert-base-arabic-camelbert-mix	Original	0.7802	0.7759	0.0043	0.0425	0.0231
	Aravec	0.8470	0.8327	0.0143	0.0187	0.0155
	CAMEL	0.8029	0.7996	0.0033	0.0215	0.0177
	Aravec_POS	0.8557	0.8434	0.0123	0.0179	0.0153
	CAMEL_POS	0.8345	0.8346	-0.0002	0.0209	0.0165
	GPT	0.9031	0.8877	0.0154	0.0185	0.0197
bert-base-multilingual-cased	Original	0.6471	0.6393	0.0079	0.0599	0.0431
	Aravec	0.7610	0.7487	0.0123	0.0470	0.0345
	CAMEL	0.7594	0.7635	-0.0041	0.0595	0.0597
	Aravec_POS	0.7698	0.7643	0.0054	0.0376	0.0286
	CAMEL_POS	0.7692	0.7535	0.0157	0.0357	0.0281
	GPT	0.8494	0.8522	-0.0028	0.0421	0.0558
MARBERT	Original	0.9788	0.9764	0.0024	0.0059	0.0065
	Aravec	0.9941	0.9928	0.0014	0.0018	0.0016
	CAMEL	0.9939	0.9936	0.0004	0.0012	0.0011
	Aravec_POS	0.9947	0.9935	0.0012	0.0013	0.0014
	CAMEL_POS	0.9945	0.9939	0.0006	0.0009	0.0010
	GPT	0.9955	0.9941	0.0014	0.0010	0.0012

Table 12: Cosine similarity statistics across all language models and prompt expansion methods in the synthetic test-set. **R** (Relevance), **NR** (Non-relevance).

C GPT prompt expansion

```
Suggest 5 subheaders for the following query: "{arabic_prompt}".
For each subheader, suggest 5 words that the user can use to write the essay.

Return the answer in the following format:
1. First subheader: [list of suggested words or terms].
2. Second subheader: [list of suggested words or terms].
3. Third subheader: ...
4. Fourth subheader: ...
5. Fifth subheader: ...
"""
```

Figure 2: GPT-4o prompts messages that have been used to expand the Arabic prompt

D SVM classification

	Embeddings				**Embeddings+SS**			
Prompt	**Original**		**GPT**		**Original**		**GPT**	
Models	**Acc**	**F1**	**Acc**	**F1**	**Acc**	**F1**	**Acc**	F1
CAMeL-Lab/bert-base-arabic-camelbert-mix	65	65	74	74	66	66	78	77
aubmindlab/bert-base-arabertv2	63	63	65	65	66	66	67	67
paraphrase-multilingual-MiniLM-L12-v2	**73**	**71**	**79**	**77**	**88**	**88**	**91**	**91**
UBC-NLP/MARBERT	67	65	77	77	68	66	79	79
UBC-NLP/ARBERT	59	57	78	77	65	63	81	81
omarelshehy/arabic-english-sts-matryoshka-v2.0	67	63	79	77	71	69	83	83
nomic-ai/nomic-embed-text-v2-moe	52	38	62	56	58	49	68	65
sentence-transformers/LaBSE	62	57	77	76	68	65	85	85
sentence-transformers/distiluse-base-multilingual-cased-v1	58	50	62	56	73	71	77	76
bert-base-multilingual-cased	60	60	65	65	62	62	66	66

Table 13: Performance of different models on synthetic test set using two input settings: (i) Embeddings: pair of prompt,essay, and (ii) Embeddings + similarity score (SS). Original and GPT-based prompts are compared. Acc and F1 in (%).

E Prompt Engineering

Prompt schema (English-translated) for small LLMs

Instruction:

You perform binary classification: is the given topic covering the given essay or not. You receive an essay and a topic as input. Return only the word `"Yes"` if the topic comprehensively covers the essay, or `"No"` if it does not. If you return any other words, you will be fined $1000.

Input:

Essay:

> My favorite day was a sunny Saturday. I spent with my family at the beach. We swam, built sandcastles, and watched the sunset together — I felt completely happy.

Topic:

> Describe your favorite day.

Does the essay comprehensively cover the topic?

Response:

Yes

Input:

Essay:

> I bought a car and I'm happy to share that with you.

Topic:

> Describe your favorite day.

Does the essay comprehensively cover the topic?

Response:

No

Input:

Essay:

> `{{essay_text}}`

Topic:

> `{{prompt_text}}`

Does the essay comprehensively cover the topic?

Response:

F Small LLM classification

Prompt	Original		GPT	
Small LLM	**Acc**	**F1**	**Acc**	**F1**
Synthetic test set				
Falcon-0.5B-Instruct	51	60	48	56
Falcon-1.5B-DeepInstruct	97	96	88	90
Qwen3-0.6B	90	91	64	73
Qwen3-1.7B	**97**	**97**	85	83
Gemma-3-1B-it	81	76	53	14
Gemma-3-4B-it	80	76	53	15
Jais-Family-590m	57	69	50	67
Jais-Family-1p3b	76	77	60	67
Jais-Family-2p7b	78	81	83	85
Jais-Family-6p7b	91	90	80	76
QAES				
Falcon-0.5B-Instruct	46	27	47	24
Falcon-1.5B-DeepInstruct	88	87	80	75
Qwen3-0.6B	63	69	48	62
Qwen3-1.7B	**90**	**89**	76	69
Gemma-3-1B-it	61	36	51	00
Gemma-3-4B-it	60	34	50	00
Jais-Family-590m	47	59	49	63
Jais-Family-1p3b	82	80	57	60
Jais-Family-2p7b	75	78	61	57
Jais-Family-6p7b	81	77	69	72
QAES + ZAEBUC				
FalconH1-0.5B-Instruct	49	21	49	17
FalconH1-1.5B-DeepInstruct	**95**	**94**	91	90
Qwen3-0.6B	76	78	57	67
Qwen3-1.7B	92	92	82	80
Gemma-3-1B-it	60	34	51	01
Gemma-3-4B-it	59	31	50	01
Jais-Family-590m	52	61	51	63
Jais-Family-1p3b	86	85	64	66
Jais-Family-2p7b	79	81	64	66
Jais-Family-6p7b	81	77	69	64

Table 14: Performance of small LLMs with Arabic support on different datasets using original and GPT-based prompts. Acc and F1 in (%).

G Setup parameters and settings

Component	Configuration / Settings
Prompt Expansion	**Word2Vec**: `full_grams_sg_300_wiki` **CAMeL-BERT**: `bert-base-arabic-camelbert-mix` **POS**: `CAMeL_BERT disambiguator` **GPT**: engine = `gpt-4o`, temperature = 0.7
SBERT Threshold	0.5
SVM (Scikit-learn)	Classifier: `SVC` (Support Vector Classifier) Parameters: kernel = "rbf"; probability = True
Falcon-H1	max_new_tokens = 3 (2 service tokens + 1 content token) temperature = 0.3; do_sample = True repetition_penalty = 1.1; top_p = 0.8; early_stopping = True
Gemma	max_new_tokens = 2; temperature = 0.3; top_p = 0.8
Qwen3	Default settings from `generation_config.json` Temperature = 0.6; TopP = 0.95; TopK = 20; MinP = 0 (Thinking mode uses the same settings; greedy decoding is avoided)
Fine-tuning	Batch size = 16; Epochs = 3 Training objective: CosineSimilarityLoss warmup_steps = 100 **Optimizer:** AdamW (`lr=2e-5`, eps=1e-6, betas=(0.9, 0.999), weight_decay=0.01) **Scheduler:** Linear learning rate decay with warmup (100 steps), final LR = 0

Table 15: Experimental setup and hyperparameter configurations.