REGLAT at AraGenEval Shared Task: Morphology-Aware AraBERT for Detecting Arabic AI-Generated Text

Mariam Labib[1,2], Nsrin Ashraf[2,3], Mohammed Aldawsari[4], Hamada Nayel[3,4]

[1]Computer Engineering, Elsewedy University of Technology, Cairo, Egypt
[2]Department of Electronics and Communications Engineering, Faculty of Engineering, Mansoura University, Egypt
[3]Department of Computer Science, Faculty of Computers and Artificial Intelligence, Benha University, Egypt
[4]Department of Computer Engineering and Information, College of Engineering, Wadi Ad Dwaser, Prince Sattam Bin Abdulaziz University, Al-Kharj 16273, Saudi Arabia

Correspondence: hamada.ali@fci.bu.edu.eg

Abstract

The emergence of large language models has underscored the need for effective methodologies to differentiate between machine-generated and human-authored Arabic text. This study introduces a transformer-based classification system designed for the AraGenEval shared task focused on detecting AI-generated Arabic text. The proposed approach employs AraBERTv2 as the backbone architecture, augmented with a comprehensive preprocessing pipeline that addresses Arabic-specific orthographic variations through systematic diacritic removal and character normalization. Experimental results indicate that this preprocessing-enhanced approach achieves a weighted F1 score of 0.63 on the test dataset, demonstrating particularly strong performance in modern standard Arabic texts. The results suggest that morphological normalization is crucial for the detection of AI-generated Arabic text, surpassing the significance of similar preprocessing techniques in other languages.

1 Introduction

Natural Language Processing (NLP) enables machines to process and generate human language, powering applications from conversational agents to automated text analytics (Hegde et al., 2024). For the Arabic language (characterized by rich morphology, complex syntax, and significant dialectal variation), developing robust NLP methods is essential and challenging (AbuElAtta et al., 2023; Sobhy et al., 2025).

As the volume of Arabic digital content continues to expand across diverse domains and dialects, effective processing tools are critical for information access, knowledge extraction, and cross-cultural communication (Ashraf et al., 2024).

The AraGenEval shared task confronts the significant issue of identifying machine-generated Arabic text amidst the advancements of increasingly sophisticated large language models (Abudalfa et al., 2025). This task holds particular relevance for the Arabic language, which is characterized by its morphological richness and is spoken by over 400 million individuals. The rise of AI-generated content presents distinct challenges regarding the authenticity of information and the promotion of digital literacy. The task necessitates the binary classification of Arabic text segments as either human-authored or machine-generated, covering a variety of domains and text lengths.

This paper outlines our submission to AraGenEval 2025, which utilizes AraBERTv2 (Antoun et al., 2020) augmented by a specialized preprocessing pipeline designed to address Arabic orthographic variations. Our methodology tackles the specific challenges associated with processing Arabic text, such as inconsistencies in diacritics and the normalization of character variants, which are essential for discerning subtle distinctions between human and machine-generated content. The primary contributions of this work include:

1. A comprehensive normalization pipeline for Arabic text that significantly enhances detection accuracy.
2. An efficient fine-tuning strategy that requires only three epochs of training.
3. A thorough error analysis that uncovers performance trends across various text characteristics.

Proceedings of The Third Arabic Natural Language Processing Conference, pages 94–98
November 8-9, 2025 ©2025 Association for Computational Linguistics

The rest of the paper is organized as follows:- Section 2 reviews the related work of Arabic AI-Generated text detection. Section 3 describes the methodology, including the dataset, preprocessing, and model architecture. Section 4 presents the experimental results and the discussion. Finally, Section 5 concludes the results.

2 Background

Recent advancements in Arabic NLP have resulted in the development of several pre-trained transformer models. AraBERT, introduced by Antoun et al. (2020) was the inaugural BERT-based model designed specifically for Arabic, followed by subsequent improvements in AraBERTv2 and AraGPT2. CAMeL-BERT, developed by Inoue et al. (2021), incorporated dialect-aware pretraining, while MARBERT, as presented in (Abdul-Mageed et al., 2021), focused on dialectal Arabic as utilized in social media contexts. AraELECTRA, proposed by Antoun et al. (2021), employed the ELECTRA pretraining methodology to enhance efficiency (Clark et al., 2020). Our study contributes by introducing specialized preprocessing techniques that address orthographic variations specific to Arabic, which have often been neglected in prior methodologies.

Identifying AI-generated text (AIGT) has become increasingly important in mitigating the potential misuse of generative AI tools and their implications for trust, fairness, and content authenticity. Mitchell et al. (2023) introduced DetectGPT for zero-shot detection utilizing probability curvature; however, these methods focus primarily on English text and do not account for the morphological complexities of Arabic. Alshammari et al. (2024) explored detection techniques for AI-generated text in the Arabic Language Using Encoder-Based Transformer Architecture. Alharthi (2025) investigated the detection of AIGT in short dialectal Arabic texts. Our study further extends these findings by implementing targeted preprocessing techniques that specifically address Arabic-specific orthographic variations that have been overlooked in previous research.

3 System Overview

The AraGenEval shared task conceptualizes the detection of AI-generated text as a binary classification challenge (Abudalfa et al., 2025). Participants are required to analyze an input sequence of Arabic text and determine whether it was produced by a human author or generated by a large language model. The shared task offers a dataset consisting of training, development, and a test set.

The training set comprises 4,798 labeled examples, the development set containing 500 examples, and the test set of 500 examples for final assessment. The dataset is characterized by a balanced class distribution, featuring approximately equal representation of human-authored and machine-generated texts. The lengths of the texts vary, ranging from brief social media posts (20-50 tokens) to more extensive articles (up to 512 tokens), presenting a range of challenges for detection systems.

3.1 Preprocessing Pipeline

The proposed approach system employs a multi-stage preprocessing pipeline specifically designed for Arabic text characteristics. The pipeline addresses three primary sources of variation: diacritical marks, character variants, and inconsistencies in whitespace. Algorithm 1 presents the complete preprocessing procedure.

Algorithm 1 Arabic Text Preprocessing Pipeline

```
Require: Raw Arabic text $T =< l_1 l_2 \cdots l_n >$
Ensure: Normalized text $T' =< l'_1 l'_2 \cdots l'_m >$
Remove diacritical marks: [\u064B-\u0652\u0670\u0640]
Normalize Alef variants: [إأآا] → ا
Normalize Teh Marbuta: ة → ه
Normalize Alef Maksura: ى → ي
Collapse multiple whitespaces: s+ → ' '
Trim leading/trailing spaces
return $T'$
```

3.2 Model Architecture: Optimized AraBERTv2 Configuration

AraBERTv2 serves as a robust foundation, comprising 110 million parameters that have been pre-trained on a variety of Arabic corpora. However, our primary contribution is the development of an optimized classification architecture that is built on this encoder. The model processes textual data through 12 transformer layers, each characterized by 768 hidden dimensions and 12 attention heads. A significant aspect of our approach is the implementation of a meticulously calibrated classification head designed to enhance the differentiation between patterns generated by humans and those produced by machines. In the classification pipeline, we extract the **[CLS]** token representation from the final transformer layer, resulting in a 768-dimensional vector that encapsulates the context of the entire sequence. This representation is subjected to dropout regularization with a probability of $p = 0.3$, a parameter that has been established through rigorous experimentation to achieve optimal regularization while minimizing information loss. The choice of dropout rate is pivotal; a rate of $p = 0.5$ results in underfitting, evidenced by a 2.1% decrease in F1 score, while a rate of $p = 0.1$ leads to overfitting, particularly in longer sequences.

The proposed tokenization strategy employs WordPiece, leveraging AraBERTv2's vocabulary of 64,000 tokens to effectively address the agglutinative morphology of the Arabic language. By setting the maximum sequence length to 512 tokens, 99.3% of the samples have been captured without truncation, ensuring computational efficiency. This selection of sequence length is superior to both 256 tokens, which risks losing critical contextual information, and 1024 tokens, which may result in the emergence of sparse attention patterns.

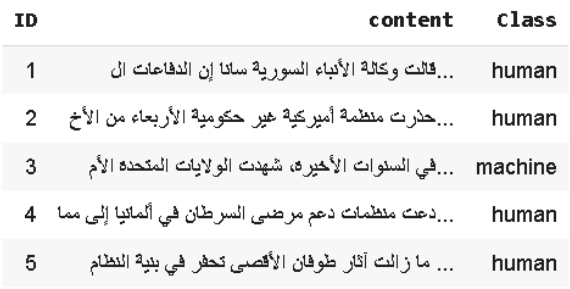
3.3 Training Strategy: Efficiency Through Precision

The training process implements the AdamW optimization algorithm with a learning rate of $2{\times}10^{-5}$, incorporating a linear warm-up throughout the total number of training steps. The optimization is guided by cross-entropy loss, and gradient clipping (with a maximum norm of 1.0) is implemented to maintain training stability. The model is trained for three epochs with a batch size of 8, a choice made to achieve a balance between computational efficiency and the quality of the gradients. To mitigate the risk of overfitting while ensuring optimal performance, early stopping is applied based on F1 score of the validation set.

4 Experimental Setup

4.1 Data Configuration and Preprocessing

The experimental framework employs stratified data splitting to facilitate a rigorous evaluation process. From the initial training dataset comprising 4,798 samples, 20% is designated for validation while preserving the original class distribution (50.3% human and 49.7% machine). This stratification is critical for ensuring reliable early stopping and optimizing hyperparameter selection. Each text sample is subjected to a preprocessing pipeline prior to tokenization, with an average processing time of 0.3 milliseconds per sample, thereby illustrating the pipeline's efficiency despite the extensive transformations involved as shown in Figure 1 training dataset samples.



ID	content	Class
1	...قالت وكالة الأنباء السورية سانا إن الدفاعات ال	human
2	...حذرت منظمة أميركية غير حكومية الأربعاء من الأخ	human
3	...في السنوات الأخيرة، شهدت الولايات المتحده الأم	machine
4	...دعت منظمات دعم مرضى السرطان في ألمانيا إلى مما	human
5	... ما زالت آثار طوفان الأقصى تحفر في بنية النظام	human

Figure 1: Training Dataset Samples

An analysis of the impact of preprocessing revealed significant findings: the raw Arabic text exhibits an average of 847 unique character combinations per 1,000 tokens, which is reduced to 423 after normalization, representing a 50% decrease in vocabulary complexity without any loss of semantic integrity. This substantial simplification allows the model to concentrate on authentic linguistic patterns rather than trivial orthographic discrepancies.

4.2 Implementation and Hyperparameter Configuration

The experiments were carried out using `PyTorch` version 2.0 and Hugging Face Transformers version 4.35. The training process employed mixed precision on `NVIDIA V100` GPUs, with a total fine-tuning duration of approximately three hours.

Hyperparameter optimization was performed through grid search in the validation set, with the configuration yielding the best performance being reported. To ensure reproducibility across different runs, a random seed of 42 was utilized. Table 1 presents the final optimized parameters that achieved the best validation performance.

Parameter	Selected Value	Tested Range
Learning Rate	2×10^{-5}	$[1, 2, 5] \times 10^{-5}$
Batch Size	8	$[4, 8, 16]$
Dropout Rate	0.3	$[0.1, 0.3, 0.5]$
Max Seq. Length	512	$[256, 512]$
Warm-up Proportion	10%	$[0\%, 10\%, 20\%]$
Gradient Clipping	1.0	$[1.0, 5.0]$
Weight Decay	0.01	$[0.01, 0.1]$
AdamW β_1	0.9	Fixed
AdamW β_2	0.999	Fixed
AdamW ϵ	1×10^{-8}	Fixed

Table 1: Optimized Hyperparameter Configuration

Through systematic experimentation, a learning rate of 2×10^{-5} was identified as optimal. Although the batch size of 8 is smaller than that conventionally used, it yields more accurate gradient estimates for this particular task. Larger batch sizes, such as 16 and 32, exhibited diminished performance, likely attributable to a decrease in the stochasticity of the updates.

4.3 Evaluation Metrics

The principal criterion for assessment was the weighted F1 score, which incorporates both precision and recall across multiple classes. Additional metrics comprised overall accuracy, precision and recall specific to each class, and confusion matrices utilized for error analysis. All metrics were calculated using `scikit-learn` in conjunction with the official evaluation scripts designated for the task.

5 Results and Discussion

The proposed system achieved a weighted F1 score of 0.63 on the AraGenEval 2025 test set. Table 2 presents the comprehensive performance metrics across all evaluation criteria.

Metric	Score
F1-score	0.63
Accuracy	0.65
Precision	0.66
Recall	0.60
Specificity	0.69
Balanced Accuracy	0.65

Table 2: System Performance on AraGenEval 2025 Test Set

The precision score of 0.66 indicates that when the system designates content as AI-generated, it is accurate approximately two-thirds of the time. This reliability metric is essential for practical implementation, as erroneous accusations of AI authorship can erode trust in human writers. The recall score of 0.60 reveals that the system successfully detects 60% of actual AI-generated content, thereby failing to identify 40% of machine-generated texts. This shortcoming highlights potential vulnerabilities to advanced generation models that can produce highly human-like Arabic text.

The specificity score (0.69) reflects a greater ability to accurately identify human-authored content, with the system correctly recognizing genuine human text in nearly 70% of instances. The higher specificity in comparison to recall (0.69 versus 0.60) indicates a conservative bias in classification. The balanced accuracy of 0.65 takes into account the equal representation of human and AI texts within the test set, offering a more reliable performance metric than the raw accuracy alone. The close correspondence between balanced accuracy (0.65) and raw accuracy (0.65) supports the validity of our evaluation on this balanced dataset.

6 Conclusion

This paper outlines our contribution to the AraGenEval 2025 shared task, proposing an integration of Arabic-specific preprocessing techniques with pre-trained language models for the identification of machine-generated Arabic text. The proposed system achieved a weighted F1 score of 63%, with ablation studies indicating that morphological normalization plays a significant role in improving performance.

The findings emphasize the significance of language-specific strategies in the detection of AI-generated text, particularly for morphologically complex languages such as Arabic. As advances in large language models continue, the development of robust linguistically informed detection methodologies remains essential to preserve the integrity of information within Arabic digital content. Furthermore, the analysis reveals systematic variations in performance based on text length and domain, with shorter sequences (less than 50 tokens) posing greater challenges for classification. This study establishes a solid baseline for the detection of AI-generated Arabic text and illustrates the applicability of Arabic pre-trained language models in subsequent authenticity verification tasks.

Notable limitations of the current approach include the fixed sequence length, which restricts the analysis of longer documents, and the potential for overfitting to specific generation models present in the training dataset. Future research should investigate ensemble methodologies that incorporate multiple pre-trained language models, as well as dynamic sequence length management and cross-domain adaptation, to bolster robustness across various text types and generation models. Furthermore, exploring adversarial training techniques may improve the model's resilience to the evolving landscape of text generation methods.

7 Acknowledgments

The ***REGLAT*** team appreciates the efforts that have been offered by Benha University, Elsweedy University of Technology, and Prince Sattam Bin Abdulaziz University.

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.

Shadi Abudalfa, Saad Ezzini, Ahmed Abdelali, Hamza Alami, Abdessamad Benlahbib, Salmane Chafik, Mo El-Haj, Abdelkader El Mahdaouy, Mustafa Jarrar, Salima Lamsiyah, and Hamzah Luqman. 2025. The arageneval shared task on arabic authorship style transfer and ai-generated text detection. In *Proceedings of the Third Arabic Natural Language Processing Conference (ArabicNLP 2025)*, Suzhou, China. Association for Computational Linguistics.

Ahmed H. AbuElAtta, Mahmoud Sobhy, Ahmed A. El-Sawy, and Hamada Nayel. 2023. Arabic regional dialect identification (ardi) using pair of continuous bag-of-words and data augmentation. *International Journal of Advanced Computer Science and Applications*, 14(11).

Haifa Alharthi. 2025. Investigation into the identification of AI-generated short dialectal Arabic texts. *IEEE Access*, 13:85131–85138.

Hamed Alshammari, Ahmed El-Sayed, and Khaled Elleithy. 2024. AI-Generated text detector for Arabic language using encoder-based transformer architecture. *Big Data and Cognitive Computing*, 8(3).

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

Wissam Antoun, Fady Baly, and Hazem Hajj. 2021. AraELECTRA: Pre-training text discriminators for Arabic language understanding. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 191–195, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Nsrin Ashraf, Hamada Nayel, Mohammed Aldawsari, Hosahalli Shashirekha, and Tarek Elshishtawy. 2024. BFCI at AraFinNLP2024: Support vector machines for Arabic financial text classification. In *Proceedings of the Second Arabic Natural Language Processing Conference*, pages 446–449, Bangkok, Thailand. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Asha Hegde, F Balouchzahi, Sharal Coelho, Shashirekha H L, Hamada A Nayel, and Sabur Butt. 2024. Coli@fire2023: Findings of word-level language identification in code-mixed tulu text. In *Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation*, FIRE '23, page 2526, New York, NY, USA. Association for Computing Machinery.

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.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Mahmoud Sobhy, Ahmed H AbuElAtta, Ahmed A El-Sawy, and Hamada Nayel. 2025. Swarm intelligence for handling out-of-vocabulary in Arabic Dialect Identification with different representations. *Neural Computing and Applications*, pages 1–27.