AraNLP at MAHED 2025 Shared Task: Using AraBERT for Text-based Hate and Hope Speech Classification

Enas A. Hakim Khalil
Systems & Information Department,
National Research Center (NRC),
Giza, Egypt
ea.khalil@nrc.sci.eg

Wafaa S. El-Kassas
Faculty of Computers and Information Technology, The National Egyptian E-Learning University (EELU), Giza, Egypt
wafaa.elkassas@gmail.com

The AraNLP system is designed for the MAHED Shared Task 2025 on text-based hate and hope classification in Arabic. The challenge was divided into three subtasks: (1) Text-based Hate and Hope Speech Classification, (2) Emotion, Offensive, and Directed Hate Detection (Multitask), and (3) Multimodal Hateful Meme Detection. The AraNLP system based on the AraBERT model achieved Macro F1-score 65% for Sub-task 1 and the results are published in the leaderboard, with rank 20. After submitting the results, the proposed model was updated to improve its performance and achieved Macro F1-score 70%, this makes the AraNLP system nearly equivalent to rank 4 in the leaderboard.

1 Introduction

Nowadays, social media platforms (e.g. Twitter, Facebook, etc.) facilitate expression of free speech. These platforms are very popular for users to have discussions and conversations and express their thoughts and opinions, but users sometimes use them to provide hate towards each other (Khezzar, Moursi, and Al Aghbari 2023). Moreover, anonymity provided to users on these platforms allows the spread of hate speech and other offensive material (Alwateer et al. 2025). Early and accurate detection of hate speech is important for maintaining a respectful and safe online environment, especially with the content's continuous and rapid growth which can lead to negative reactions from users (Al-Saqqa 2024). Therefore, there is an essential need to automatically detect and report occurrence of hate speech in text for different languages. In recent years, researchers focus on analyzing data shared on social media platforms but their attention is mainly directed towards the content written in English (Fat'hAlalim et al. 2025). In contrast, other languages such as Arabic needs more attention from researchers and needs more resources that facilitate the research tasks to get better results. Focusing on Arabic NLP tasks brings many challenges such as the lack of Arabic language resources, difficult grammatical structure, dialectal variations, and human annotation errors (Abdelsamie, Azab, and Hefny 2026). The goal of the MAHED 2025 Text-based Hate and Hope Speech Classification (Sub-task 1) is to classify Arabic text as hate, hope, or not_applicable. Such challenge is very important to encourage the research community to focus more on the tasks related to the Arabic content. The proposed AraNLP system uses the AraBERT model (Antoun, Baly, and Hajj 2020) and the experimental results demonstrate that the model achieves promising results for Sub-task 1.

2 Related Work

Hate speech is a very challenging and complex task especially with Arabic dialects as previous studies have often used multiple Arabic dialects combined in a single dataset without identifying the dialects used, which is challenging because it can lead to misidentification of non-hateful and hateful contexts related to a particular dialect (Abdelsamie et al. 2026). Moreover, the lack of adequate research on Arabic dialects and the lack of large, publicly available datasets highlight the need for more investigations about the Arabic hate speech detection (Fat'hAlalim et al. 2025).

Many studies about the detection of hate speech in Arabic tweets or social media posts in general have used different methods such as machine learning techniques, deep learning, the application of transfer learning, Arabic BERT-based models, and the Large Language Models (LLMs) (Al-Saqqa, Awajan, and Hammo 2024). In addition, researchers have examined hybrid models that combine different approaches to propose ensemble methods that incorporate multiple deep learning techniques to improve results (Al-Saqqa et al. 2024).

Proceedings of The Third Arabic Natural Language Processing Conference, pages 706–711

The arHateDetector Framework was proposed to detect hate speech in the Arabic tweets by (Khezzar et al. 2023). The authors conducted several experiments to evaluate machine learning algorithms like logistic regression, Linear SVC, and Random Forest, in addition to deep learning models like AraBERT and Convolutional Neural Networks (CNNs). The experiments prove that AraBERT outperformed the other models achieving the best performance across seven different datasets.

An interpretable framework to detect hate speech in Arabic was developed based on LLMs by (Alwateer et al. 2025). The authors focus to enhance understanding of model decisions by combining interpretable machine learning methods with advanced Natural Language Processing (NLP) techniques in their proposed method. The results show the effectiveness of combining LLM with interpretability to provide a transparent solution for automated detection of harmful content.

In (Fat'hAlalim et al. 2025), the authors analyze Arabic hate speech detection using advanced transformer-based models across three datasets collected from different social media platforms. The analysis includes the effects of data augmentation, oversampling, and model interpretability using the LIME method. The monolingual transformer-based models achieved significant performance improvements. Besides, they applied cross-validation across datasets to evaluate the generalization capabilities of models.

In (Abdelsamie et al. 2026), the authors focused on understanding of dialect-specific hate speech and proposed a multi-task learning approach built upon transformer architecture to bridge the gap in hate speech detection across Arabic dialects. They used publicly available datasets from various dialects, the proposed model was designed to identify and differentiate subtle hate speech patterns and use shared representation knowledge across five Arabic dialects: Egyptian, Gulf, Saudi, Levant, and Algerian.

While deep learning, transfer learning, and ensemble learning approaches have shown potential, many challenges persist, specifically with Arabic language difficulties and dialectal variations (Fat'hAlalim et al. 2025). In (Ramos et al. 2024), the authors highlights that Transformer models consistently outperform other methods, but their high computational requirements suggest that hybrid approaches, combining deep learning with traditional machine learning, may be more suitable in certain contexts.

Although significant steps have been made in addressing low-resource languages like Arabic, there is still a need for further research work to improve inclusivity across a wider range of cultural and linguistic contexts (Ramos et al. 2024).

3 Data

The dataset (Zaghouani, et al., 2024) (Biswas & Zaghouani, 2025) (Biswas & Zaghouani, 2025) used in the MAHED 2025 Sub-task 1 consists of Arabic text (MSA and dialect) with train file of 6893 tweets, validation file of 1476 tweets, and test file of 1477 tweets. Figure 1 shows the detailed data format consisting of: tweet id, data and label. Each tweet is classified with one of three labels: hate, hope, or not_applicable.

Table 1 illustrates examples of classified tweets for different labels. The statistics of the dataset in terms of the number of tweets per label is provided in Table 2. It can be observed that the dataset is imbalanced in the count of tweets of the represented labels and this note explains later why the trained model is biased towards the not_applicable label.

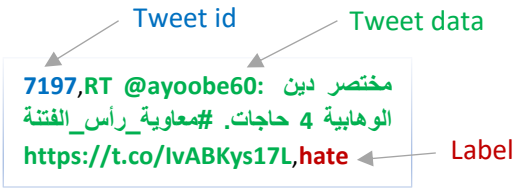

Figure 1: Train data format.

Tweet	Label
كل المهاجرين لصوص ومجرمون يجب طردهم فوراً	Hate
معاً يمكننا بناء مستقبل أفضل لأطفالنا	Hope
اليوم هو يوم مشمس وجميل	not_applicable

Table 1: Examples of different classes of labeled data.

Label	Train	Validation	Test
hope	1892	409	422
hate	1304	262	287
not_applicable	3697	806	768
(All Labels)	6893	1476	1477

Table 2: Statistics of the dataset.

4 Methodology

The proposed system mainly uses the AraBERT model (Antoun et al. 2020). AraBERT uses the BERT-base configuration that has 12 encoder blocks, 768 hidden dimensions, 12 attention heads, 512 maximum sequence length, and a total of 110M parameters. AraBERT has different versions, and all models are available in the HuggungFace model page under the aubmind name. In the proposed system, the twitter AraBERT: 'aubmindlab/bert-base-arabertv02-twitter' model is used (HuggingFace n.d.). This model is selected because it was pretrained on 60M Arabic tweets with different dialects and Modern Standard Arabic (MSA) (i.e. it is a general Arabic or a simplified version that does not use diacritics and it is usually used in newspapers, tweets, etc.) which is similar in nature to the MAHED 2025 Sub-task 1 dataset. Besides, Arabertv02-twitter has better vocabulary coverage for slang, hashtags, and emojis.

Figure 2 shows the proposed system based on AraBERT model. The AraBERT Preprocessor (ArabertPreprocessor) is used for the training and validation files to clean the Arabic text. This step is important for handling the unique characteristics of the Arabic language, such as diacritics and ligatures. Then, the data is tokenized using the AraBERT tokenizer (AutoTokenizer) to convert the text into numerical tokens. After that, the distribution of sentence lengths is analyzed to help determine an appropriate maximum length. The maximum sentence length is determined to be 128 and truncate longer sentences to avoid performance degradation. 27, and 5 tweets of all the training, and validation tweets respectively exceed the maximum length and have been truncated.

Figure 2 shows the **text classification model** using the fine-tuning approach on a pre-trained **AraBERT model** (aubmindlab/bert-base-arabertv02-twitter). The training is processed with the training parameters such as learning rate, batch size, etc. Table 3 illustrates the used uniform hyper parameter settings for AraBERTv02-twitter.

Once the training is complete, the final fine-tuned model is saved to be used later in the prediction phase. The last step is to predict the output labels of new and unseen text data in the test set using the saved model with predict_labels and classification report libraries.

All experiments are carried out on the Google Colab environment and covers the entire machine learning pipeline from data preparation to model training and evaluation. The Google Colab platform is used with a NVIDIA L4 GPU, System RAM 6.6 / 53.0 GB, GPU RAM 1.3 / 22.5 GB, and Disk 40.7 / 235.7 GB.

Parameter	Value
adam_epsilon	1e-8
learning_rate	2 e -5
Number of train epochs	2
warmup_ratio	0
per_device_train_batch_size	16
per_device_eval_batch_size	128
gradient_accumulation_steps	2
do_eval	True
load_best_model_at_end	True
metric_for_best_model	'macro_f1'
greater_is_better	True
Seed	25

Table 3: Hyper parameters for AraBERTv02-twitter.

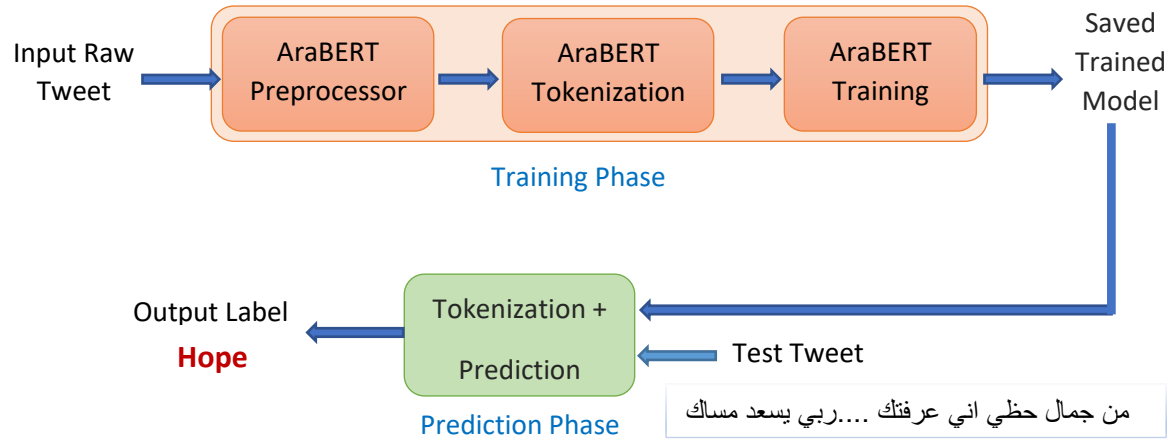

Figure 2: AraBERT Tweet Classification Model with labels hope, hate, or not applicable.

Epoch	Training Loss	Validation Loss	Macro F1	Accuracy	Macro Precision	Macro Recall
1	No log	0.667	0.650	0.680	0.668	0.637
2	No log	0.670	0.659	0.675	0.656	0.662

Table 4: Training Log Table.

5 Results

The performance of proposed system is evaluated on the validation set after each training epoch. The used evaluation metrics include **accuracy**, **macro F1-score**, **precision**, and **recall**, along with a **confusion matrix**.

The training results are recorded in table 4. Although the validation Loss (Model error on validation data) is slightly increased from 0.667 to 0.670 through the two epochs, which might indicate no improvement or slight overfitting but Macro F1 went from 0.650 to 0.659, which means the model got a little better. The Accuracy is dropped slightly from 0.680 to 0.675, also do Macro Precision from 0.668 to 0.656 meanwhile Macro Recall improved from 0.637 to 0.662.

The confusion matrix for training process shown in figure 3, the Class 0 (not_applicable) has approximately 19% errors, mostly confused with class 1 (hope) while class 1 got about 43% errors, mostly confused with class 0 and class 2 (hate) got 48% errors mostly confused with class 0. The model heavily counts on toward predicting class 0 when unsure.

To test the model, several experiments have been done. So, to differentiate between these experiments, they are referenced in this paper as Test 1, Test 2, and Test 3. Test 1 is the first

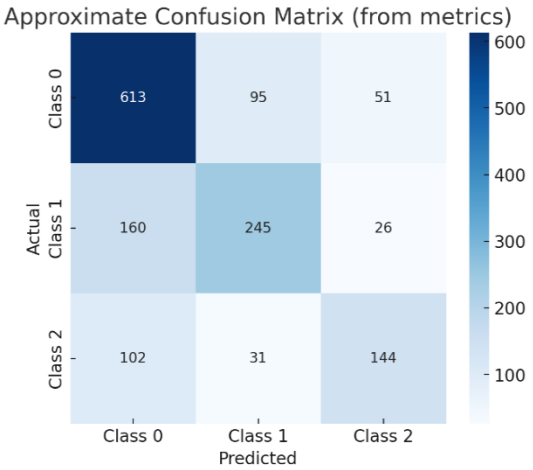

Figure 3: Confusion matrix for regular training process.

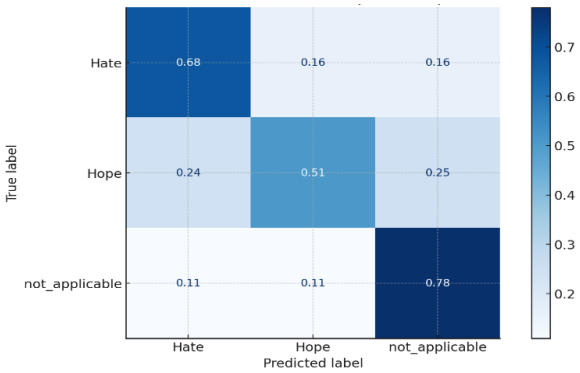

Figure 4: Normalized Confusion Matrix for Test 2.

experiment, and its results are published through competition in the leaderboard. Test 2 provides improved results than Test 1. The difference in results for the two tests comes from the predict_label library that was used in Test 2 instead of the **pipeline** in Test 1 for getting output labels. Test 2 is more accurate and represents better results.

From the classification report for test data (Test 2 results) in Table 5, it can be observed that while 68% of real hate is correctly predicted only 51% of real hope tweets are correctly found. 78% of not_applicable tweets are correctly predicted. The corresponding confusion matrix in figure 4 shows that about half of hope actual class tweets are confused with other labels, which confirms with the low recall results for hope class. These results suggest that the model generalize well because no performance drop from validation to test which suggests minimal overfitting.

In the third experiment (Test 3), both train and validation data were added in a single file called trainall.txt and this file was used to train the model with 5 fold Cross Validation and the number of epochs was increased from 2 (default value) to 5.

	Precision	Recall	F1-score	Support
Hate	0.70	0.68	0.69	287
Hope	0.69	0.51	0.59	422
not_applicable	0.68	0.78	0.72	768
Accuracy			0.68	1477
Macro Avg	0.69	0.66	0.67	1477
Weighted Avg	0.68	0.68	0.68	1477

Table 5: Classification Report for Test 2 (on Test Set).

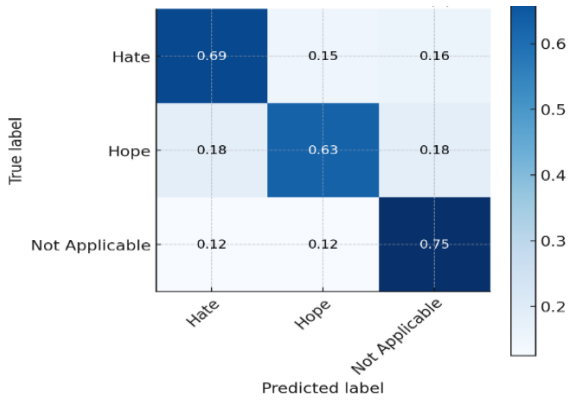

Figure 5: Normalized Confusion Matrix for Test 3.

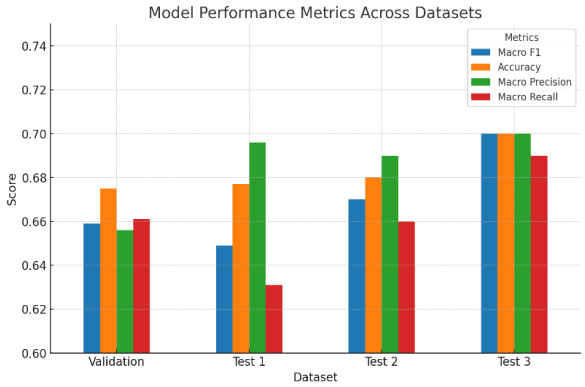

Figure 6: Validation vs. Test Results Comparison.

These modifications improve the Test 3 results over all metrics compared to Test 1 and Test 2.

From the classification report for Test 3 (the **last improved results**) in Table 6, it can be observed that while 69% of real hate tweets is correctly predicted and only 63% of real hope tweets are correctly found. 75% of not_applicable tweets are correctly predicted. The corresponding confusion matrix for Test 3 in Figure 5 shows that the best class not_applicable (recall = 0.75) and the weakest one is hope class. The results suggest that the model generalize well because no performance drop from validation to test which suggests minimal overfitting.

Table 7 and Figure 6 compare the validation results for the training process with the predicted output labels for the test data using the saved trained model. The three experiments results for test data are compared in Table 7. The difference between the three experiments for test data have been explained earlier in this section.

In the three test experiments, the model generalizes well with no performance drop from Validation to Test, which suggests minimal overfitting. It is clear that augmenting more data samples in training helps to climb higher in performance in the experiment Test 3 which achieves the better results compared to Test 1 and Test 2 results.

	Precision	**Recall**	**F1-score**	**Support**
Hate	0.70	0.69	0.69	287
Hope	0.70	0.63	0.66	422
not_applicable	0.71	0.75	0.7	768
Accuracy			0.70	1477
Macro Avg	0.70	0.69	0.70	1477
Weighted Avg	0.70	0.69	0.70	1477

Table 6: Classification Report for Test 3 (on Test Set).

	Macro F1	**Accuracy**	**Macro Precision**	**Macro Recall**
Validation	0.659	0.675	0.6561	0.661
Test 1 (Leaderboard)	0.649	0.677	0.696	0.631
Test 2	0.67	0.68	0.69	0.66
Test 3	**0.70**	**0.70**	**0.70**	**0.69**

Table 7: Results of Validation and Test data.

6 Conclusion

Recently, the detection of hate speech from social media such as Twitter gains attention of researchers. Real-time detection of harmful content is essential to safeguarding vulnerable communities, so it becomes essential to make continued research and development in Arabic hate speech detection.

This paper focuses on the domain of detecting hate and hope speech in Arabic. AraBERT model is used to detect hate and hope speech in Arabic Tweets. Three different experiments have been done on the test data and the results are compared and explained. The evaluation of AraNLP system shows promising and better results in Test 3 than Test 1 and Test 2.

Future work includes evaluating the proposed system on various hate speech datasets to evaluate the performance of both the multilingual and monolingual models. Also, oversampling techniques can be used to address the class imbalance to improve the proposed model performance. In addition, conducting extensive experiments by using and evaluating different transformer-based models and Large Language Models (LLMs) to achieve better results.

References

Abdelsamie, Mahmoud Mohamed, Shahira Shaaban Azab, and Hesham A. Hefny. 2026. "The Dialects Gap: A Multi-Task Learning Approach for Enhancing Hate Speech Detection in Arabic Dialects." *Expert Systems with Applications* 295:128584. doi:https://doi.org/10.1016/j.eswa.2025.128584.

Al-Saqqa, Samar. 2024. "Hate Speech Detection of Arabic Text Using Deep Learning and Transfer Learning Models." PhD Thesis, Princess Sumaya University for Technology (Jordan).

Al-Saqqa, Samar, Arafat Awajan, and Bassam Hammo. 2024. "A Survey of Hate Speech Detection for Arabic Social Media: Methods and Datasets." *Procedia Computer Science* 251:224–31. doi:https://doi.org/10.1016/j.procs.2024.11.104.

Alwateer, M., I. Gad, M. Elmarhomy, G. Elmarhomy, H. Hashim, M. Almaliki, and E. S. Atlam. 2025. "Interpretable Arabic Hate Speech Detection Using Large Language Model." Pp. 1–8 in *2025 2nd International Conference on Advanced Innovations in Smart Cities (ICAISC)*.

Antoun, Wissam, Fady Baly, and Hazem Hajj. 2020. "AraBERT: Transformer-Based Model for Arabic Language Understanding." Pp. 9–15 in *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, edited by H. Al-Khalifa, W. Magdy, K. Darwish, T. Elsayed, and H. Mubarak. Marseille, France: European Language Resource Association.

Biswas, Md. Rafiul, and Wajdi Zaghouani. 2025a. "An Annotated Corpus of Arabic Tweets for Hate Speech Analysis." *CoRR*. https://arxiv.org/abs/2505.11969.

Biswas, Md. Rafiul, and Wajdi Zaghouani. 2025b. "EmoHopeSpeech: An Annotated Dataset of Emotions and Hope Speech in English and Arabic." *CoRR* abs/2505.11959. https://arxiv.org/abs/2505.11959.

Fat'hAlalim, Ahmed, Yongjian Liu, Qing Xie, and Nahla Ibrahim. 2025. "Advancements in Transformer-Based Models for Enhanced Hate Speech Detection in Arabic: Addressing Dialectal Variations and Cross-Platform Challenges." *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 24(8). doi:10.1145/3748492.

HuggingFace. n.d. "Bert-Base-Arabertv02-Twitter." Retrieved September 12, 2025. https://huggingface.co/aubmindlab/bert-base-arabertv02-twitter.

Khezzar, Ramzi, Abdelrahman Moursi, and Zaher Al Aghbari. 2023. "ArHateDetector: Detection of Hate Speech from Standard and Dialectal Arabic Tweets." *Discover Internet of Things* 3(1):1. doi:10.1007/s43926-023-00030-9.

Ramos, Gil, Fernando Batista, Ricardo Ribeiro, Pedro Fialho, Sérgio Moro, António Fonseca, Rita Guerra, Paula Carvalho, Catarina Marques, and Cláudia Silva. 2024. "A Comprehensive Review on Automatic Hate Speech Detection in the Age of the Transformer." *Social Network Analysis and Mining* 14(1):204. doi:10.1007/s13278-024-01361-3.

Zaghouani, Wajdi, Md Rafiul Biswas, Mabrouka Bessghaier, Shimaa Ibrahim, Georgio Mikros, Abul Hasnat, and Firoj Alam. 2025. "MAHED Shared Task: Multimodal Detection of Hope and Hate Emotions in Arabic Content." in *Proceedings of the Third Arabic Natural Language Processing Conference (ArabicNLP 2025)*. Suzhou, China: Association for Computational Linguistics.

Zaghouani, Wajdi, Hamdy Mubarak, and Md. Rafiul Biswas. 2024. "So Hateful! Building a Multi-Label Hate Speech Annotated Arabic Dataset." Pp. 15044–55 in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. Torino, Italia: ELRA and ICCL.