AUT Journal of Modeling and Simulation

AUT J. Model. Simul., 57(2) (2025) 127-138
DOI: 10.22060/miscj.2025.24262.5414

Efficient Arabic Hate Speech Detection via LLaMA-3: A Prompting and Instruction-Tuning Approach

Anas Khudhur Abbass[1], Heshaam Faili[2]*

[1] Alborz Campus, University of Tehran, Tehran, Iran
[2] School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran, Iran.

ABSTRACT: Cyberspace generates user-generated content daily, but also gives freedom of expression, potentially spreading hate speech and endangering minorities. So, there's the issue of identifying hate speech as quickly as possible to stop it from being shared. This is particularly challenging for Arabic given its rich morphology and lack of good quality linguistic resources. In this article, we examine zero-shot and few-shot prompting for detecting Arab hate speech using the LLaMA-3-8B language model while also refining performance via supervised fine-tuning utilizing a custom instruction-based dataset. In the zero-shot approach, the outputs from the model are unstructured textual outputs, so we take the unstructured responses and run them through a lightweight TF-IDF + Logistic Regression classifier to classify the responses in one of the predefined hate speech categories. To obtain better classification, we construct the instruction-based training set by creating tweet embeddings from Arabic-BERT and using K-Means clustering to enforce semantic/topical variety. Next, we use the GPT-4o model to generate representative instructions from each cluster and create an instruction-based fine-tuning data set. We then fine-tune LLaMA-3-8B using QLoRA, which also allows the model to be fine-tuned with a lower memory footprint. The experimental results presented in this paper show that zero-shot and few-shot prompting achieved relatively low F1-scores of 42.2% and 45.0%, respectively, and instruction-tuning fine-tuning achieves the overall performance of an F1-score of 90.1%, which exceeds stronger benchmarks like AraBERT. Our results exemplify the potential impact of instruction tuning and QLoRA-based fine-tuning over prompting-based approaches in low-resource contexts like Arabic.

Review History:

Received: Jul. 13, 2025
Revised: Oct. 27, 2025
Accepted: Oct. 29, 2025
Available Online: Dec. 31, 2025

Keywords:

Hate Speech Detection in Arabic
Zero-Shot Prompting
Few-shot Prompting
Supervised Fine-Tuning
LLaMA-3-8B

1- Introduction

Social media platforms are spaces for protest, activism, and the exchange of information, serving as crucial weapons for citizens and governments alike. While they do extend the reach of persons, they also streamline the spread of harmful material such as hate speech, pornography, and cyber abusers [1, 2]. Hate Speech (HS) is defined as any communication that is derogatory towards a person or group, based on certain characteristics. Studies have revealed a direct correlation between Internet hate speech and an increase in real-world hate crimes [3].

People often use offensive language when addressing opposing viewpoints, and given the wide visibility and reach of social media, this behavior can significantly impact public perception and deepen social divides [4]. Hate speech is injurious to individuals, in producing pain and fear, and also to society by increasing social hostilities, by developing conditions which are conducive to hostilities, and ultimately to violence. The extension of the HS group to offensive and derogatory speech towards any group degrades social cohesion, which provides conditions for marginalization of groups whose numbers may be limited, and also, in extreme cases, produces real-life violence and hate crimes. It is obvious, therefore, that curbing the growth of hate speech would have a critical bearing on the viability of diverse, inclusive societies [5].

Research in the early days of the subject of detection of hate speech was centered on English, which has ready access to documentation [6]. However, recent efforts have expanded to under-resourced languages like Arabic [7]. Corresponding to this, a marked increase in the social hostilities index in Arab nations has been noted [8]. Even though Arabic is the official language in all Arab-speaking nations, generally, people use a variety of dialects in the daily interchange of thought. Each country, and even sections of countries, has dialects of its own, which makes it difficult to process Arabic text concerning the contents of the different types of social media. Arabic, being morphologically a very rich language, presents additional difficulties due to the distinction existing between the Modern Standard Arabic (MSA) - Dialectical Arabic (DA). This variation in dialects will present difficulties to the Natural Language Processing (NLP) people [9]. The

*Corresponding author's email: hfaili@ut.ac.ir

difficulty of dealing with the subject in the Arabic language dialects is very important because of the lack of the same information to work upon, for the reason that the dialects are so diverse that it makes it very difficult to do anything to develop the proper NLP [10]. Of course, the MSA is used in formal communications, and dialects of the different countries are used almost exclusively in matters of social media, for it is the common traits and the other distinct traits that make the process difficult as a whole [11].

Large Language Models (LLMs) are very powerful tools that can understand and produce human languages with great accuracy [12]. In addition, these large kind of LLMs are trained on various texts so that they may learn to show the patterns of language and the nuances of language [13]. The use of LLMs can be effective in detecting hate speech. Research has shown that these models can analyze Arabic text to identify hateful and offensive language. By pinpointing such speech, LLMs play a crucial role in reducing the prevalence of hate speech. This impact is particularly noticeable on social media platforms, where hate speech can spread rapidly [14]. An important advantage of the implementation of LLMs for hate speech detection in the Arabic language lies in the ability of these Languages models to understand the inherent complexity of natural language. The Arabic language, with it's linguistic richness and diversity, offers unique challenges and difficulties of effective detection of hate speech by traditional methods [15]. LLMs excel in processing Arabic text and capturing its intricacies such as dialectal variations, cultural references, and colloquial expressions [16]. In addition, LLMs can develop with the dynamic aspects of hate speech as an example. The various tactics or patterns of language change with the passage of time, therefore the LLMs themselves could be updated and reeducated to keep their effectiveness in against the developing threats of hate speech [17].

This paper includes key contributions to the study of Arabic hate speech detection:

- We conduct a variety of evaluations on zero-shot and few-shot prompting of the LLaMA-3-8B model and establish a baseline of performance for LLMs in low-resource Arabic dialect applications. In the zero-shot approach, a Term Frequency-Inverse Document Frequency (TF-IDF) classifier + Logistic Regression (LR) is used to classify the model outputs.
- Additionally, we demonstrate an efficient and scalable fine-tuning pipeline due to the creation of a semantically diverse instruction-based dataset (using Arabic-BERT to generate embeddings and K-Means to cluster) and by implementing an instruction-creation procedure based on each cluster using GPT-4o. The instruction tuning enhancements made to the dataset allow for effective instruction tuning of LLaMA-3-8B, followed by resource-efficient fine-tuning using Quantized Low-Rank Adaptation (QLoRA) on a performant high-performance model.

The rest of this paper is structured as follows: In Section 2, we examine related work on hate speech detection and describe prior methods from traditional machine learning and deep learning perspectives. The proposed method consists of zero-shot and few-shot prompting, development of the instruction-based dataset, instruction generation, and fine-tuning via QLoRA as described in Section 3. Section 4 offers our experimental results that were generated from various prompting and fine-tuning strategies. Section 5 provides our discussion of what we learnt and emphasizes the strength of our proposed method and how it compares to existing metrics. Finally, Section 6 concludes the paper and provides possible directions for future work.

2- Related Works

In the initial phases of research on hate speech detection, the research on NLP was almost entirely English-based, largely due to the abundance of annotated datasets and resources available. However, the development of Arabic was blocked in its infancy due to the absence of labeled corpora specific to hate speech. The first studies in Arabic NLP were in the areas of sentiment analysis and opinion mining [18], but did not include hate speech.

Hate speech detection, sentiment analysis, and emotion detection have multiple methodological aspects in common. Sentiment analysis is the process of categorizing the polarity of a given text, that is, whether the expressed opinion is in a positive or negative context. In contrast, emotion detection tries to classify particular emotions which are mentioned in the text as anger or fear. Hate speech detection, however, aims to detect harmful or discriminatory comments toward an individual or group [19, 20].

The first work done in Arabic to address hate speech was lexicon-based, where a list of hate-related words was employed to classify the content of various types of media, employing techniques derived from sentiment analysis. Mubarak et al. [21] employed these lexicons of offensive terms to measure their effectiveness in identifying hate speech in social media. Although these methods were simple, they faced limitations due to a lack of comprehensive, dialect-aware lexicons, which frequently made it challenging to detect subtle or implicit expressions of hate speech.

As the field matured, researchers began adding classical machine learning techniques to improve detection performance. Commonly used classifiers were Support Vector Machines (SVM), Naive Bayes (NB), and LR, which were trained on features derived from text (n-grams, TF-IDFs, and syntactic patterns). Aljarah et al. [22] pointed out that it is difficult to detect, given the complexity of this language, the large number of dialects of Arabic. They wanted to build an intelligent detection system of cyber hate speech languages using machine learning and natural language processing terminology. The authors annotated a set of 3,696 Arabic tweets on issues such as sport, religion, etc. The tweets were classified into three types of tweets: hate, non-hate, and neutral. There were 843 hate tweets and 790 non-hate tweets used to build the model. For the detection of the indicators in the tweets, some features of the user profile and methods such as text vectorization were added. The study applied machine

learning models such as SVM, NB, Decision Tree (DT), and Random Forest (RF). The best results were obtained for the random forest model, giving 91.3 percent accuracy using TF-IDF and the user profile features.

However, these models also encountered problems, although an advance over the lexicon techniques, owing to the linguistic richness, morphological complexity, and regional dialectic diversity of the Arabic language. The advent of deep learning systems, in particular Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), which can learn automatically the hierarchical features of the text, marks a new beginning. Anezi [23] demonstrated that deep learning uniquely met the demands of the Arabic language and reported superior accuracy than earlier studies and traditional methods. A new dataset is presented that includes 4203 Arabic comments. These were carefully coded into seven classes: anti-religion content, racist content, hate-speech relative to gender, violent or offensive comments, threatening or bullying comments, and normal positive and negative comments. The comments were classified into seven classes using the techniques of deep learning and specifically using deep bidirectional RNNs (DRNNs). There were two suggested architectures: DRNN-1 with five layers, and DRNN-2 with ten layers. Mohaouchane et al. [24] used a dataset of 15,050 Arabic YouTube comments from polemical videos with Arab stars. For each comment the several annotators from Arabic-speaking countries indicated whether comments were offensive or not to reflect some of the dialectal differences. The authors used four different architectures of neural networks in the experiment: CNN, Bidirectional Long Short-Term Network (Bi-LSTM), Bi-LSTM with attention, and a combined CNN-LSTM model. The CNN-LSTM network was best able to detect offensive comments. Salameh et al. [25] emphasized the Levant dialects, which possess an especially complex linguistic structure. They investigated the influence of different techniques of word embedding, AraVec, AraBERT, and ArabGlossBERT, in deep learning techniques, via the Jordanian Hate Speech Corpus (JHSC), comprising 400,000 annotated tweets. The embeddings were combined with LSTM and CNN deep learning techniques, and SMOTE was employed to balance the data, thereby employing an attention layer to enhance contextual understanding. It was found that ArabGlossBERT proved superior to the other embeddings, achieving 60% accuracy employing LSTM architecture, and 52% accuracy with the CNN.

Such a finding is indeed promising and signifies a substantial improvement over earlier studies and traditional methods. Deep learning models had outperformed the earlier methods, but sometimes there would be limitations in the way that the usefulness of the deep learning outputs could be used, which would often hinder the potential that might be related to the existence of large annotated Arabic datasets. The growth of pre-trained language models, designed specifically with this in mind for Arabic, viz, AraBERT [26], MARBERT [27], and Arabic BERT [28], has been a turning point in relation to the work on hate speech detection. Such models have been trained by means of transfer learning on large Arabic corpora. This has enabled the models to acquire knowledge of the contextual, semantic, and dialectical relations within Arabic speech. Studies using these models have shown widespread improvements to be made to classification accuracy.

Khezzar et al. [29] addressed the issue of Arabic hate speech detection on Twitter, noting the particular need for both Standard and dialectal forms of Arabic to be assessed. This resulted in the proposition of arHateDetector, a system based on the premise of a new dataset compiled for the purpose, namely arHateDataset, which contains about 34,000 tweets, taken from numerous Arabic dialects, of which 32% were labeled as hate speech. A thorough form of pre-processing was undertaken, and different machine learning and deep learning models were compared: these included LinearSVC, CNN, and AraBERT, the last of which showed the best performance. Daouadi et al. [30] proposed a Contextual Deep Random Forest (CDRF) method to classify Arabic tweets into hate and non-hate classes. They utilized two datasets for their Bayesian model. The first dataset consisted of 11634 tweets, and the second dataset was composed of 5340 tweets. The CDRF method comprised steps such as preprocessing, contextual embedding, a multi-grained scanning step, and a cascade forest step. According to the results, the CDRF method showed improved results in comparison with 14 baseline algorithms. Alwateer et al. [31] made use of a survey to classify 9352 tweets. These tweets were classified as normal (62.45%), abusive (29.56%), and hate (8%). They used Arabic BERT and CAMeLBERT, which were learnt with learning objectives adjusted for the three-class classification problem.

Recently, LLMs and instruction-tuned models that can be conditioned on a natural language prompt and that are usable without task-specific fine-tuning are receiving attention. For example, ChatGPT [32], LLaMA models, etc., are being used, and recently, “few-shot” or even “zero-shot” prompting is being applied to allow the end user to specify hate speech detection on new domains and tasks in general with less data. Das et al. [33] applied ChatGPT to the detection of hate speech in 11 languages. They employed three separate datasets: HateCheck (3,728 test cases in English), Multilingual HateCheck (36,582 test cases over 10 languages), and HatemojCheck (3,930 test cases with emoji). Their methods were functional testing on the GPT-3.5-turbo model functions, applied to a binary classification prompt (hate speech vs. non-hate speech) and a standard definition of hate speech, followed by evaluations of performance on accuracy and macro F1 scores. Pan et al. [34] investigated the detection of harassment and sexism on social media, especially with LLMs, through zero-shot learning, few-shot learning, and fine-tuning in a comparative study. The EDOS dataset was utilised (which consists of English data containing 20,000 examples of sexism data from Reddit and Gab), and the HatEval dataset, which is of almost 13,000 English tweets containing the data for hate speech detection. The methodology consisted of fine-tuning both encoder-only models (BERT, DeBERTa) and encoder-decoder models (Zephyr, Mistral) with a classification layer, utilising LoRA,

Table 1. The L-HSAB dataset splitting.

Split	Abusive	Normal	Hate	Total
Train	1,369	2,968	339	4,676
Test	359	682	129	1,170

Arabic Hate Speech Detection
Given a tweet in the Arabic language, determine its category.
Categories:
- Abusive: Contains offensive or insulting language.
- Normal: Contains no offensive or harmful language.
- Hate: Promotes hate speech or incites violence against individuals or groups.
Instruction: Your output should be one of the following labels: `abusive`, `normal`, or `hate`.
Input Tweet: `{TEXT}`
Prediction:

Fig. 1. Zero-Shot Prompt.

context learning through zero-shot learning, using 5-shot and 5-shot with Sentence Transformers. This shift could be of particular use to low-resourced languages, such as Arabic, because it lessens the direct dependence on large annotated corpora and opens the doors to further progress in the identification of hate speech concerning a more nuanced appreciation of context.

3- Methodology

This study aims to develop a robust model for detecting hate speech in Arabic tweets by leveraging the LLaMA-3-8B language model [35]. The proposed methodology consists of three main components:

- Prompting-based classification, including both zero-shot and few-shot prompting strategies.
- Instruction dataset generation, which involves clustering semantically similar tweets and generating corresponding task instructions.
- A TF-IDF + LR classification layer, which maps the unstructured textual outputs of LLaMA-3-8B into discrete labels (abusive, normal, or hate) by transforming the responses into numerical feature vectors and applying supervised classification.
- Supervised fine-tuning using QLoRA, enabling efficient model adaptation with reduced computational cost.

3- 1- Dataset

The L-HSAB dataset [36, 37], consists of Arabic tweets labeled into three classes: abusive, normal, and hate. Table 1 indicates that the training data consists of 4,676 tweets (1,369 abusive, 2,968 normal, and 339 hate), and the test data has 1,170 tweets (359 abusive, 682 normal, and 129 hate). The L-HSAB dataset was used for all experiments, so results are comparable across all models.

3- 2- Prompting-Based Classification

We conducted the initial stage by assessing the performance of the LLaMA-3-8B model on hate speech detection in a few-shot and zero-shot prompting context. In the zero-shot prompting, the LLaMA-3-8B model was provided with a textual instruction and a single input tweet in the model output-response phase, as shown in Fig. 1. The model generated an unstructured textual response describing the classification of the tweet (e.g., “This tweet is abusive”). In this stage, the text response generated by the LLaMA-3-8B model was first converted into a numerical feature vector using TF-IDF. This feature vector was then fed to an LR model to automatically classify it into one of three classes: abusive, normal, or hate.

In the few-shot, the prompt, which is shown in Fig. 2, included multiple examples with labels to help the model

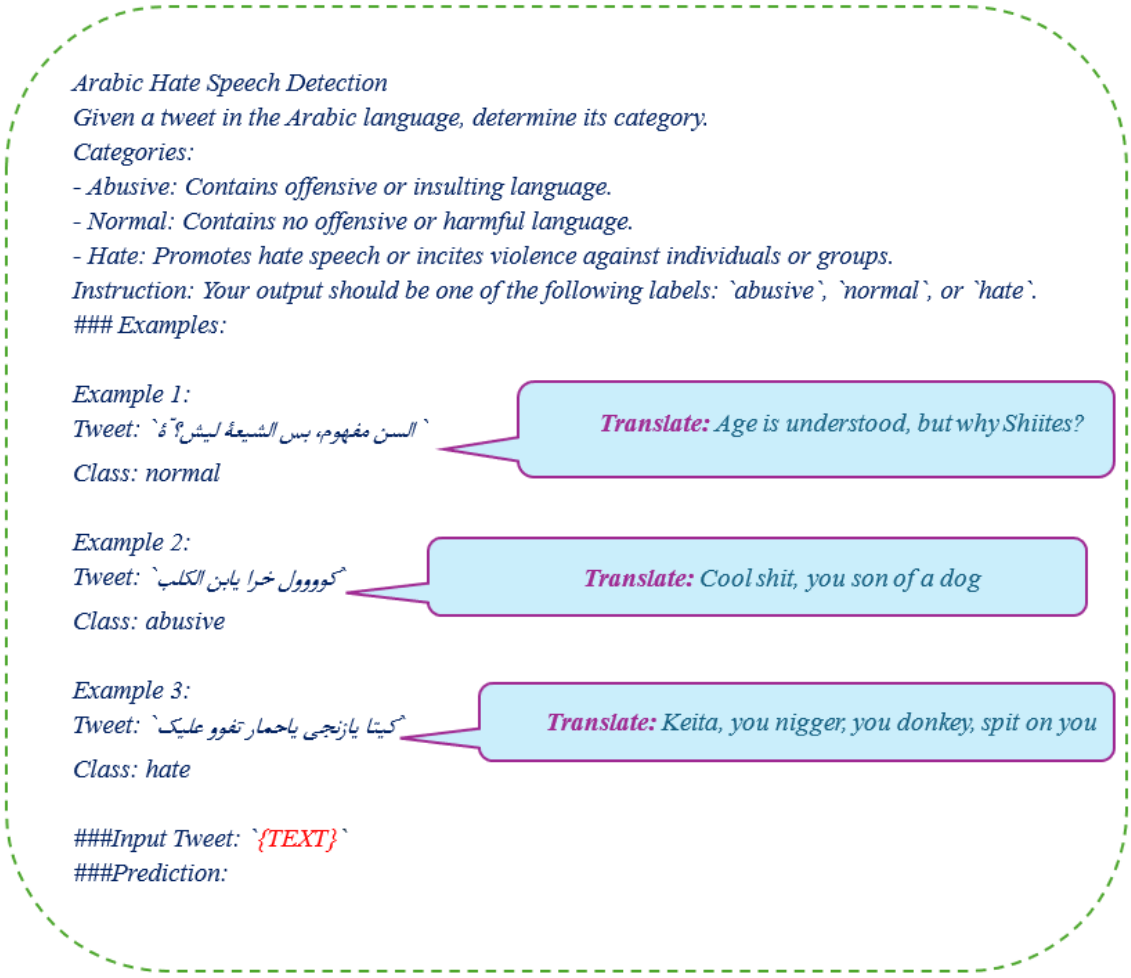


Fig. 2. Few-Shot Prompt.

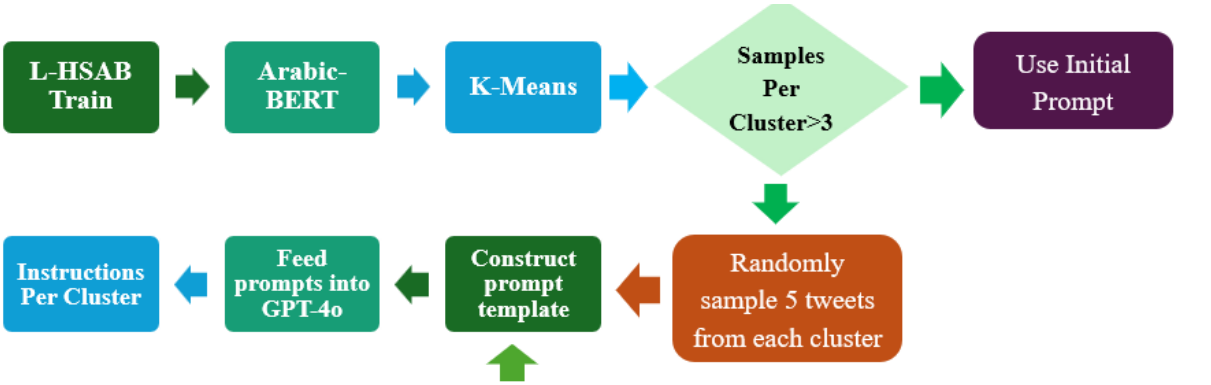
generalize its output to the classification task. The model performed better in this setup because it relied on the task context to provide guidance.

3- 3- Instruction Dataset Generation

The instruction dataset was made to improve the model's ability to detect hate speech through a multi-step approach illustrated in Fig 3. The instruction dataset used in this approach is a means of simulating user behavior and increases the training dataset while breaking the class imbalance the dataset had and exposing the model to a broader set of use cases or patterns in Arabic hate speech. The steps to complete instruction dataset generation instruction were as follows:

1) **Vectorization with Arabic BERT:** Each tweet from the training set was encoded into a dense vector representation using the Arabic-BERT model, with a semantic property as a component of the Sentence Transformer framework.
2) **Clustering:** K-Means clustering was applied to the tweet embeddings to group semantically similar tweets. To find a suitable number of clusters, we examined the variation of the Inertia metric (as defined in Formula (1)), which gave the measurement of the internal homogeneity of clusters. Not as clearly illustrated in Fig. 4, the Inertia value drops sharply until about 20 clusters, where it shows significant improvement in cluster compactness. After this number of clusters, the decrease is much less rapid, for example, from 0 to 20 clusters, the Inertia goes down by about 70,000 units, whereas from 20 to 40 clusters the decrease is about 10,000 units or so. This slight improvement suggests that, past 20 clusters, the clusters do not provide meaningful benefits. K = 20 was selected as the number of clusters to utilize, ensuring a balance between cluster granularity and semantic coherence. Each of the clusters produced represents tweets related to individual topics or linguistic properties.

$$\text{Inertia} = \sum_{k=1}^{K} \sum_{x \in c_k} \|x - \mu_k\|^2 \quad (1)$$



Arabic Hate Speech Detection Instruction Generation Task:
Analyze the following Arabic tweets and write prompt templates per cluster that can guide a LLM to classify tweets into the following categories.

- **Abusive:** Contains offensive or insulting language.
- **Normal:** Contains no offensive or harmful language.
- **Hate:** Promotes hate speech or incites violence against individuals or groups.

Consider that:

- Your prompt templates for clusters should be different.
- No more than 150 words.
- Make your output as a dictionary of {"cluster-id": "prompt template"}
- Set of examples will be provided for you to use them to get task nature.

Examples for the Arabic hate speech are:
{cluster_id}

Fig. 3. Instruction Generation steps.

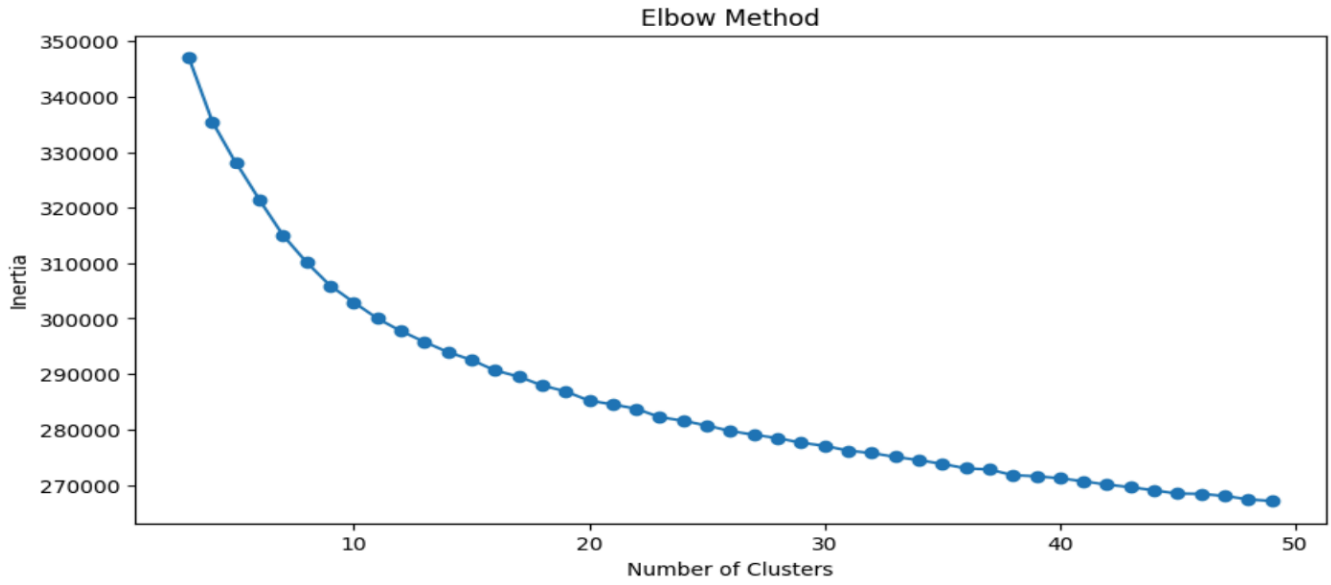


Fig. 4. The results for determining the optimal number of clusters.

Table 2. An example of generated instructions for the first 3 clusters.

Cluster 0	Classify the following tweet into one of the three categories: 'Abusive' for offensive or insulting language, 'Normal' for neutral language with no harmful content, or 'Hate' for language promoting hate speech or inciting violence.
Cluster 1	Determine whether the tweet contains harmful or offensive language. Label as 'Abusive' for insults, 'Normal' for neutral content, and 'Hate' for any expression of hate or incitement to violence.
Cluster 2	Please classify this tweet into one of the categories: 'Abusive' for offensive or derogatory remarks, 'Normal' for general conversation with no offensive intent, or 'Hate' for harmful or violent rhetoric.

Where is total number of clusters, is cluster , is data point in the cluster, is centroid of the cluster . Inertia reflects the compactness of clusters, with lower values indicating tighter, more cohesive clusters.

3) **Filtering:** We sample to have fair and proper representative coverage over the clusters. If a cluster consists of more than three samples, we randomly select five tweets from the cluster and produce instructions for maximizing instructional diversity. For clusters of three or less, we include all tweets from these clusters and produce instructions once per sample to ensure equal representation of small clusters.

4) **Prompt Template Development and Instruction Generation:** We employed a systematic template to construct prompts that guide the LLM in generating instructions (the prompt is presented in Fig 3). The prompt contains 5 distinct samples per cluster, representative of cluster data, as an example for generating prompts. Later, the constructed prompt is fed into GPT-4o to generate instructions per cluster. Each instruction was designed to guide a model in effectively identifying and classifying hate speech in Arabic tweets. Table 2 is a sample of the generated guidelines for clusters one through three, designed to identify and classify hate speech in Arabic tweets.

In this implementation, 20 clusters were determined. In Table 3, the "Cluster" column shows the cluster number (from cluster-0 to cluster-19) that each tweet was placed in based on semantic similarity, using the K-Means algorithm. Some clusters perform better (e.g., cluster-4 with an accuracy of 82.0%), while others (e.g., cluster-3 with an accuracy of 35.4%) show weaker performance. This difference may be due to the diversity in linguistic patterns or the semantic complexity of the tweets in each cluster. For example, the tweets in cluster-4 may have clearer linguistic features for identification.

3- 4- Supervised Fine-tuning

We fine-tune the LLaMA-3-8B model using QLoRA, a parameter-efficient technique requiring much less memory during training. Two fine-tuning settings are used, one with the original training data with regular prompts (Supervised Fine-Tuning (SFT)) and one also using enriched instruction-based data (SFT + INST). Training uses multiple epochs of fine-tuning data with some hyperparameters adjusted for lower-resource hardware (e.g., less GPU memory). Overall, this allows for scalable model adaptation and requires no loss of classification accuracy.

3- 4- 1- SFT Without Instructions

The SFT process used the zero-shot prompt template as described in 3.2 Section. QLoRA was implemented to relieve memory constraints. The QLoRA parameters were ranked *r=8, lora_alpha=16*, and *lora_dropout=0.05*. The SFT was trained for 20 epochs with a learning rate of with the *AdamW* 32-bit optimizer, with a batch size of *2*. The SFT completed in 23 hours on an NVIDIA RTX 3090 GPU with 24GB memory, with 4 CPUs with 50GB memory.

3- 4- 2- SFT with Instructions (SFT + INST)

To remedy the class imbalance in the dataset, the instruction dataset was used to augment the training set. The clusters with high F1-scores for abusive (clusters 4, 14) and hate (clusters 4, 6, 7, 9, 11, 12, 16) classes were selected to make samples to balance (closer to the normal class) the dataset. Same QLoRA was applied, but SFT was trained for 10 epochs. All models used zero-shot prompting for testing to ensure a fair and balanced comparison across models.

4- Results

All experiments were implemented on NVIDIA H100 GPUs with 80 GB of memory. Running time for the complete model was approximately 12–16 hours. The step of generation of instructions, which may also be performed by a CPU (but

Table 3. Performance metrics of Tweet clustering.

Cluster	Abusive (F1-Score)	Hate (F1-Score)	Normal (F1-Score)	Accuracy	Macro Precision	Macro Recall	Macro F1-Score
cluster-0	68.6	37.8	88.5	76.8	65.4	64.9	64.9
cluster-1	67.3	41.6	83.9	72.3	63.5	67.6	64.3
cluster-2	59.4	27.4	68.8	61.4	62.9	54.2	51.9
cluster-3	48.4	16.7	15.1	35.4	52.4	38.5	26.7
cluster-4	76.1	52.4	90.5	82.0	75.2	72.3	73.0
cluster-5	56.4	39.0	65.3	58.4	59.6	57.0	53.6
cluster-6	57.3	43.3	74.9	64.0	58.8	62.4	58.5
cluster-7	63.9	49.7	74.0	67.3	68.9	64.3	62.5
cluster-8	37.1	4.06	62.0	48.1	34.2	34.5	34.4
cluster-9	59.4	45.1	81.5	68.8	61.3	65.6	62.0
cluster-10	54.3	16.7	47.7	49.3	61.2	46.1	39.6
cluster-11	52.0	45.2	70.1	59.3	57.6	62.2	55.8
cluster-12	29.4	12.0	52.0	40.7	33.4	31.0	31.1
cluster-13	41.8	31.7	44.5	41.2	43.7	45.7	39.3
cluster-14	72.1	42.6	87.2	77.8	70.1	67.1	67.3
cluster-15	49.0	37.1	58.7	51.0	52.4	55.9	48.3
cluster-16	56.7	43.0	61.5	57.6	64.4	56.3	53.7
cluster-17	47.7	22.2	37.3	40.4	45.2	42.4	35.8
cluster-18	41.7	37.9	84.5	66.6	55.5	57.9	54.7
cluster-19	52.4	41.7	80.7	65.5	59.1	64.3	58.3

in more time), usually takes about an hour to be executed.

The models are tested based on the test set of the L-HSAB for hate speech detection on Arabic tweets. Four methods were tested with the LLaMA-3-8B model: zero-shot prompting, few-shot prompting, SFT, and SFT + INST. The evaluation measures were accuracy, precision, recall, and F1-score. We compare the performance with two established baselines, AraBERT [38] and Multilingual BERT [39] as shown in Table 4. The zero-shot prompting approach produced the least performance outcome with an F1-score of 42.2% and an accuracy of 50.0%. This reflects LLaMA-3-8B's inability to capture the key linguistic features aligned with Arabic hate speech in the absence of task-specific training in the zero-shot prompt scenario. The few-shot prompting included three labeled examples and yielded a slight improvement, with an F1-score of 45.0% and an accuracy of 58.2%. This would suggest that context-based examples help the model identify hate speech over language. Unfortunately, both prompting methods witnessed lower performance than the baseline models, at a numerical distance. Supervised fine-tuning produced a step change in the performance outcome. The SFT method with QLoRA at 20 epochs produced an F1-score of 86.8% and accuracy at 91.9%, very close to AraBERT (F1-score: 87.0%). The SFT + INST model with instruction-augmented dataset, at 10 epochs, achieved a F1-score of 90.1%, accuracy of 92.6%, precision of 93.6%, and recall of 87.7%, surpassing all model variations and AraBERT and Multilingual BERT. This indicates that instruction generation on fine-tuning can produce effective model performance on hate speech text classification tasks. The inclusion of the instruction-augmented dataset also resolved the class imbalance of the L-HSAB by leveraging high-performing clusters (i.e., abusive - clusters 4 and 14 and hate - clusters 4, 6, 7, 9, 11, 12, and 16).

5- Discussion

SFT + INST surpassed the other methods for several significant reasons. First, the instruction dataset generation

Table 4. Performance comparison of hate speech detection models on the L-HSAB test set.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
AraBERT [38]	88.0	87.0	87.0	87.0
Multilingual BERT [39]	81.0	81.0	79.0	80.0
Zero-Shot Prompting	50.0	49.6	52.3	50.9
Few-Shot Prompting	58.2	57.2	51.7	54.3
SFT	91.9	90.3	84.9	87.5
SFT + INST	**92.3**	**93.6**	**87.7**	**90.6**

pipeline, which used Arabic-BERT embeddings and K-Means clustering (for both semantic grouping of tweets and generating instruction variety through GPT-4o), allowed the model to generalize over the complex linguistic and context facets of Arabic tweets. Second, through the QLoRA method of fine-tuning, SFT + INST was extremely memory efficient on just one NVIDIA RTX 3090 GPU. QLoRA allowed for the model to save a lot of memory without losing its ability to learn task-based semantics, and it took almost half the amount of total epochs for SFT + INST to converge (10) compared to SFT (20). In comparison to AraBERT and Multilingual BERT, SFT + INST benefited from training a large-scale pretrained model (LLaMA-3-8B) that already had better representation of a low-resource language (for example, Arabic) internally (fine-tuning) through just its pre-trained embeddings. Although AraBERT has achieved a strong F1 score of 87.0%, its performance is limited due to reliance on static embeddings. This approach may not reflect the nuances of dynamic contextual as well as the capabilities of LLaMA-3-8B's transformer architecture. Multilingual-BERT clearly performed worse (80.0 F1-score) than AraBERT, and this could be due to this model tying too many languages together (English and Arabic, for example) and thus diminishing its coherence for a single language like Arabic.

The results also demonstrate the limitations of prompting-based approaches in low-resource scenarios. The performance of zero-shot and few-shot prompting suggests that task-specific training is required for inputting dialectal and cultural nuances into Arabic hate speech detection. Nevertheless, these methods can be useful for prototyping at speed or when labeled data is limited. The variability among cluster performance shows that some hate speech is more recognizable (e.g., cluster 4), maybe because of the presence of prominent language identifiers. Meanwhile, other varieties of hate speech (e.g., cluster 3) seem harder to identify because their articulation relies on vague or contextual expression.

6- Conclusion

This study provides an approach to hate speech detection in Arabic tweets with a robust framework for hate speech detection, resulting in state-of-the-art performance on the L-HSAB dataset with the LLaMA-3-8B language model. Specifically, our SFT + INST approach is composed of instruction dataset development, supervised fine-tuning with QLoRA, and comparing zero-shot and few-shot task based prompting to produce a F1-score of 90.1% over benchmark performance with AraBERT (87.0%) and Multilingual BERT (80.0%). The instruction generation pipeline, combined with Arabic-BERT, K-Means clustering, and GPT-4o, successfully mitigated class imbalance and increased the training data richness, and QLoRA made it computationally affordable to fine-tune the model. These results highlight the promise of adapting large language models to customized learning data to address challenging natural language processing tasks for low-resource languages. The proposed approach provides a scalable solution for social media moderation in Arabic-speaking societies. In the future, the instruction dataset might be expanded to include additional dialects, and multimodal features, such as images, can be embedded into the model for input in the form of emojis, which could enhance the overall Arabic NLP model by transferring the model into other NLP tasks, like sentiment analysis or misinformation identification.

References

[1] J.Q. Dong, C.-H. Yang, Business value of big data analytics: A systems-theoretic approach and empirical test, Information & Management, 57(1) (2020) 103124.

[2] N.A. Ghani, S. Hamid, I.A.T. Hashem, E. Ahmed, Social media big data analytics: A survey, Computers in Human behavior, 101 (2019) 417-428.

[3] K. Müller, C. Schwarz, Fanning the flames of hate: Social

media and hate crime, Journal of the European Economic Association, 19(4) (2021) 2131-2167.

[4] A. Al-Hassan, H. Al-Dossari, Detection of hate speech in social networks: a survey on multilingual corpus, in: 6th international conference on computer science and information technology, ACM, 2019, pp. 10-5121.

[5] N. Badri, F. Kboubi, A. Habacha Chaibi, Abusive and Hate speech Classification in Arabic Text Using Pre-trained Language Models and Data Augmentation, ACM Transactions on Asian and Low-Resource Language Information Processing, (2024).

[6] N. Badri, F. Kboubi, A.H. Chaibi, Combining fasttext and glove word embedding for offensive and hate speech text detection, Procedia Computer Science, 207 (2022) 769-778.

[7] Z. Boulouard, M. Ouaissa, M. Ouaissa, Machine learning for hate speech detection in arabic social media, in: Computational Intelligence in Recent Communication Networks, Springer, 2022, pp. 147-162.

[8] N. Albadi, M. Kurdi, S. Mishra, Are they our brothers? analysis and detection of religious hate speech in the arabic twittersphere, in: 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE, 2018, pp. 69-76.

[9] K. Darwish, W. Magdy, A. Mourad, Language processing for arabic microblog retrieval, in: Proceedings of the 21st ACM international conference on Information and knowledge management, 2012, pp. 2427-2430.

[10] Y. Matrane, F. Benabbou, N. Sael, A systematic literature review of Arabic dialect sentiment analysis, Journal of King Saud University-Computer and Information Sciences, 35(6) (2023) 101570.

[11] M. Hedhli, F. Kboubi, Cnn-bilstm model for arabic dialect identification, in: International Conference on Computational Collective Intelligence, Springer, 2023, pp. 213-225.

[12] J.D.M.-W.C. Kenton, L.K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of naacL-HLT, Minneapolis, Minnesota, 2019, pp. 2.

[13] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners, OpenAI blog, 1(8) (2019) 9.

[14] R. Alshaalan, H. Al-Khalifa, Hate speech detection in saudi twittersphere: A deep learning approach, in: Proceedings of the fifth Arabic natural language processing workshop, 2020, pp. 12-23.

[15] B. Alrashidi, A. Jamal, A. Alkhathlan, Abusive content detection in arabic tweets using multi-task learning and transformer-based models, Applied Sciences, 13(10) (2023) 5825.

[16] M. Ibrahim, CUFE at NADI 2024 shared task: Fine-Tuning Llama-3 To Translate From Arabic Dialects To Modern Standard Arabic, in: Proceedings of The Second Arabic Natural Language Processing Conference, 2024, pp. 769-773.

[17] M.S. Jahan, M. Oussalah, D.R. Beddia, N. Arhab, A Comprehensive Study on NLP Data Augmentation for Hate Speech Detection: Legacy Methods, BERT, and LLMs, arXiv preprint arXiv:2404.00303, (2024).

[18] H. ElSahar, S.R. El-Beltagy, Building large arabic multi-domain resources for sentiment analysis, in: International conference on intelligent text processing and computational linguistics, Springer, 2015, pp. 23-34.

[19] P. Nandwani, R. Verma, A review on sentiment analysis and emotion detection from text, Soc Netw Anal Min, 11(1) (2021) 81.

[20] F.M. Plaza-Del-Arco, M.D. Molina-González, L.A. Ureña-López, M.T. Martín-Valdivia, A multi-task learning approach to hate speech detection leveraging sentiment analysis, IEEE Access, 9 (2021) 112478-112489.

[21] H. Mubarak, K. Darwish, W. Magdy, Abusive language detection on Arabic social media, in: Proceedings of the first workshop on abusive language online, 2017, pp. 52-56.

[22] I. Aljarah, M. Habib, N. Hijazi, H. Faris, R. Qaddoura, B. Hammo, M. Abushariah, M. Alfawareh, Intelligent detection of hate speech in Arabic social network: A machine learning approach, Journal of Information Science, 47(4) (2021) 483-501.

[23] F.Y.A. Anezi, Arabic Hate Speech Detection Using Deep Recurrent Neural Networks, Applied Sciences, 12(12) (2022) 6010.

[24] H. Mohaouchane, A. Mourhir, N.S. Nikolov, Detecting offensive language on arabic social media using deep learning, in: 2019 sixth international conference on social networks analysis, management and security (SNAMS), IEEE, 2019, pp. 466-471.

[25] K. Salameh, S. Hamza, S. Atiani, Enhancing Arabic Hate Speech Detection: The Role of Word Embedding Techniques with Deep Learning Models, in: 2025 International Conference on New Trends in Computing Sciences (ICTCS), 2025, pp. 334-341.

[26] W. Antoun, F. Baly, H. Hajj, Arabert: Transformer-based model for arabic language understanding, arXiv preprint arXiv:2003.00104, (2020).

[27] M. Abdul-Mageed, A. Elmadany, E.M.B. Nagoudi, ARBERT & MARBERT: Deep bidirectional transformers for Arabic, arXiv preprint arXiv:2101.01785, (2020).

[28] A.S. Alammary, BERT models for Arabic text classification: a systematic review, Applied Sciences, 12(11) (2022) 5720.

[29] R. Khezzar, A. Moursi, Z. Al Aghbari, arHateDetector: detection of hate speech from standard and dialectal Arabic Tweets, Discover Internet of Things, 3(1) (2023) 1.

[30] K.E. Daouadi, Y. Boualleg, O. Guehairia, Deep Random

Forest and AraBert for Hate Speech Detection from Arabic Tweets, J. Univers. Comput. Sci., 29(11) (2023) 1319-1335.

[31] M. Alwateer, I. Gad, M. Elmarhomy, G. Elmarhomy, H. Hashim, M. Almaliki, E.-S. Atlam, Interpretable Arabic Hate Speech Detection using Large Language Model, in: 2025 2nd International Conference on Advanced Innovations in Smart Cities (ICAISC), IEEE, 2025, pp. 1-8.

[32] P. Welsby, B.M. Cheung, ChatGPT, in, Oxford University Press, 2023, pp. 1047-1048.

[33] M. Das, S.K. Pandey, A. Mukherjee, Evaluating ChatGPT against functionality tests for hate speech detection, in: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), 2024, pp. 6370-6380.

[34] R. Pan, J.A. García-Díaz, R. Valencia-García, Comparing Fine-Tuning, Zero and Few-Shot Strategies with Large Language Models in Hate Speech Detection in English, CMES-Computer Modeling in Engineering & Sciences, 140(3) (2024).

[35] AI-Meta, meta-llama/Meta-Llama-3-8B-Instruct, (2024).

[36] H. Mulki, H. Haddad, C.B. Ali, H. Alshabani, L-hsab: A levantine twitter dataset for hate speech and abusive language, in: Proceedings of the third workshop on abusive language online, 2019, pp. 111-118.

[37] H.a.H. Mulki, Hatem and Ali, Chedi Bechikh and Alshabani, Halima, L-HSAB dataset, (2019).

[38] F. Husain, O. Uzuner, Transfer Learning Across Arabic Dialects for Offensive Language Detection, in: 2022 International Conference on Asian Language Processing (IALP), 2022, pp. 196-205.

[39] H. Al-Jarrah, M. Al-Smadi, M. Hammad, F. Shannaq, Using Deep Learning Techniques to Detect Hate and Abusive Language in Arabic Tweets, International Journal of Intelligent Engineering & Systems, 17(5) (2024).

HOW TO CITE THIS ARTICLE

A. Khudhur Abbass, H. Faili, Efficient Arabic Hate Speech Detection via LLaMA-3: A Prompting and Instruction-Tuning Approach, AUT J. Model. Simul., 57(2) (2025) 127-138.

DOI: 10.22060/miscj.2025.24262.5414