Benchmarking Large Language Models for Hate Speech Detection in Arabic Dialects: Focus on the Saudi Dialects

Omaima Fallatah
Department of Data Science-College of Computing, Umm Al-Qura University, Makkah, Saudi Arabia

***Abstract*—This study investigates the effectiveness of large language models (LLMs) in detecting Arabic hate speech, with a particular focus on prompt-based learning and the sociolinguistic challenges of Saudi dialects. We evaluate four LLMs, GPT-4o, LLaMA3, Gemma2, and ALLaM, using zero-shot, one-shot, and three-shot prompting strategies. The results show that all models benefit from in-context examples, with GPT-4o achieving the highest overall performance across all prompting settings. A detailed error analysis reveals persistent challenges, particularly in detecting implicit hate, handling dialectal variation, and interpreting culturally embedded expressions. We also highlight limitations related to topic bias and annotation ambiguity, which further complicate model evaluation. Overall, the findings offer key insights for evaluating LLMs in low-resource settings and addressing the unique linguistic complexities of Arabic dialects.**

***Keywords*—Arabic hate speech detection; large language models (LLMs); in-context learning; Arabic NLP**

I. Introduction

Over the past decade, social media platforms have become a popular space for people to share their opinions on a wide range of topics, including social and political issues. However, the ease of access and sense of anonymity have also contributed to a rise in harmful and offensive content. Hate speech, such as cyberbullying, racism, and and discriminatory language, is now a common issue across many platforms. Detecting such content automatically is essential for mitigating its harmful societal impact, yet remains a technically and culturally complex challenge. Consequently, research has focused on developing automated approaches for hate speech detection across various languages, including Arabic [1], [2].

Hate speech can be defined as language that provokes hatred or violence against individuals or groups based on identity factors such as race, religion, nationality, or gender [3]. Hate speech frequently includes dehumanizing expressions, exclusionary remarks, or implicit forms of hostility. Although significant progress has been made in hate speech detection, the task remains difficult, especially in low-resource languages and culturally diverse environments.

Arabic is the official language of 22 countries, including Saudi Arabia, and is spoken by more than 400 million people [4]. It introduces unique linguistic and sociocultural challenges to automated hate speech detection. Arabic includes both Modern Standard Arabic (MSA) and a broad spectrum of regional dialects, many of which are underrepresented in NLP datasets and resources [5], [6]. Moreover, social media content is often written in dialects, making it harder for models trained on MSA to generalize effectively [7]. Further complications arise from the frequent use of sarcasm, code-switching, and culturally specific references that affect both annotation quality and model accuracy. Therefore, our study focuses specifically on Saudi dialects to reduce dialectal variation and better capture the linguistic characteristics of a single regional context.

With the rise of Large Language Models (LLMs), such as OpenAI's GPT [8] and Google's Gemini [9], there has been an increasing interest in using these models for different NLP tasks. In the Arabic context, LLMs have shown promising performance in sentiment analysis, translation [10], text summarization [11], and punctuation prediction [12]. These models excel particularly in few-shot settings through in-context learning, where tasks are performed by providing the model with representative examples given as part of the input prompt rather than through fine-tuning.

Although progress has been made, the effectiveness of LLMs in detecting Arabic hate speech, especially in regions with significant dialectal variation such as Saudi Arabia, has yet to be thoroughly examined. According to a recent survey [1], most prior work has focused on traditional machine learning or fine-tuned models like BERT [13], while the potential of in-context learning remains largely underexplored. In contrast to BERT-based fine-tuning, which relies on large annotated datasets, prompt-based LLMs use zero- and few-shot adaptability, reducing annotation demands and improving generalization across Arabic dialects. This study therefore aims to: 1) evaluate state-of-the-art LLMs on Arabic hate speech detection using few-shot prompting, 2) conduct a comparative analysis across balanced and unbalanced datasets, and 3) perform an in-depth error analysis focused on sociolinguistic and dialectal factors in Saudi social media discourse. Beyond benchmarking, these contributions provide practical insights for dataset creation, annotation standards, and moderation systems, supporting the development of more culturally informed and effective hate speech detection tools.

The rest of this paper is organized as follows. Section II reviews related work on Arabic hate speech detection and recent developments in LLMs. Section III describes the proposed method and dataset used. Section IV presents and compares the evaluation results of the different models. Section V offers an in-depth error analysis highlighting key linguistic and contextual challenges. Finally, Section VI concludes the paper with key findings and suggestions for future work.

II. Related Work

Many studies have addressed Arabic hate speech detection, primarily relying on supervised learning with annotated datasets in MSA and various dialects. Majority of early approaches employed traditional machine learning classifiers [2], while more recent efforts have explored deep learning and transformer-based architectures, particularly AraBERT and its variants [14]. However, limitation related to dialectal coverage and annotation consistency remain a challenge. A recent review [1] highlights key trends in Arabic hate speech research particularly on Twitter, noting that most available datasets are in MSA, with dialectal Arabic still significantly underrepresented.

Several Arabic hate speech datasets have been introduced in recent years, reflecting both the linguistic and social diversity of the region. A well-established resource is ADHAR, a multi-dialectal hate speech corpus in Arabic [6], which provides annotations across various hate speech categories, including those related to nationality, religion, ethnicity, and race. In addition, [15] introduced an Arabic hate speech dataset by leveraging emojis to guide data collection. This method enabled the creation of a corpus rich in offensive and hateful content that might otherwise be difficult to detect using traditional keyword filtering. Another large-scale dataset is the Arabic Hate Speech Superset [16], which aggregates multiple public Arabic hate speech datasets, totaling over 400,000 annotated posts from platforms using dialects such as Gulf, Egyptian, and Levantine. Other corpora target specific dialects, including the Jordanian Hate Speech Corpus [17] and the Saudi Hate Speech Detection Dataset [18].

Large Language Models (LLMs) are a type of pre-trained language model built using transformer architectures and trained on massive text corpora [19]. While all LLMs are pre-trained models, the term LLM typically refers to models with billions of parameters such as as GPT [8] and Gemini [9]. In contrast, earlier or smaller-scale pre-trained models like AraBERT [14] or MARBERT [20] are often fine-tuned on specific tasks and domains, especially in low-resource languages like Arabic. LLMs such as GPT variants have significantly improved the performance of different natural language processing tasks. These models contain billions of parameters, and can perform a wide range of generation tasks, including translation, summarization, and question answering. One of the core techniques that enables LLMs to perform on such tasks without additional training is in-context learning, also referred to as prompt-based learning. In this approach, the model is guided to complete a task through instructions or some examples provided directly in the input prompt without the need for fine-tuning [21].

In the last few years, many of these models have become accessible through public APIs, making them widely usable for both academic and industrial applications. In this study, we evaluate several LLMs including OpenAI's GPT-4o [22], Meta's LLaMA3 [23], Google's Gemma2 [24], and ALLaM. The later is an Arabic-English model developed by SDAIA [25]. While GPT-4o is known for its strong multilingual capabilities, LLaMA3 and Gemma2 are open-source models optimized for efficiency and generalization. ALLaM, on the other hand, is designed with a focus on Arabic language tasks, aiming to better handle the linguistic characteristics of Arabic text.

As a result of their growing availability, LLMs have been used in different Arabic NLP tasks. For example, [10] evaluated several LLMs, including LLaMA and Gemma, on Arabic sentiment analysis and machine translation in zero- and few-shot settings. Their findings indicated that while LLaMA demonstrated strong contextual understanding, it still underperformed compared to pretrained ArabBERT [14]. Text summarization is another NLP task where LLMs have shown promising capabilities. For example, [11] introduced a dataset for Arabic summarization consisting of article–summary pairs generated using GPT-3.5 Turbo. Another study on Arabic sentiment analysis and machine translation reported that LLMs outperform traditional machine learning models(e.g. SVM and Logistic Regression) [26], yet still lag behind state-of-the-art models in some tasks, reinforcing the value of task-specific fine-tuning [27].

In the domain of Arabic hate speech detection, the use of LLMs and pre-trained language models has shown promising results. Large pre-trained models such as AraBERT and its variants have demonstrated substantial performance improvement when fine-tuned on multi-dialectal datasets, outperforming traditional and deep learning baselines in terms of accuracy and F1-score [6], [28], [15]. For instance, on ADHAR dataset, fine-tuned models have achieved micro-averaged F1-scores above 95% on multi-label hate speech tasks and up to 94% on binary-label datasets. However, performance remains inconsistent across dialects and hate speech categories, highlighting the importance of further research into dialectal robustness and cultural nuance [15]. Moreover, given the linguistic diversity across the Arab region, effective Arabic NLP, particularly for tasks like hate speech detection, requires models that are capable of handling dialectal variation. This is especially important in regions like Saudi Arabia, where dialectal influence extends across Gulf and neighboring linguistic varieties.

In summary, while recent advances in Arabic hate speech detection have leveraged fine-tuned models and dialect-specific datasets, the role of in-context learning with LLMs remains underexplored. This study addresses that gap by evaluating the performance of several LLMs on Arabic hate speech detection using prompt-based learning. We also investigate sociolinguistic sources of error, emphasizing the role of dialectal variation and cultural context in shaping model behavior.

III. Method

Our methodology is developed to investigate the capabilities of LLMs to detect Arabic hate speech in Saudi Twitter content. The implemented approach illustrated in Fig. 1 begins by filtering the dataset and selecting the appropriate instances. After preprocessing and balancing the dataset, we apply different LLMs using multiple settings: zero-shot, one-shot, and few-shot. Finally, the models' predictions are compared against ground truth labels for evaluation.

A. Dataset

This study uses a subset of the Arabic Hate Speech Superset [16], which is publicly available via Hugging Face[1]. This dataset combines multiple publicly available Arabic hate

[1]https://huggingface.co/datasets/manueltonneau/arabic-hate-speech-superset accessed 10 May 2025

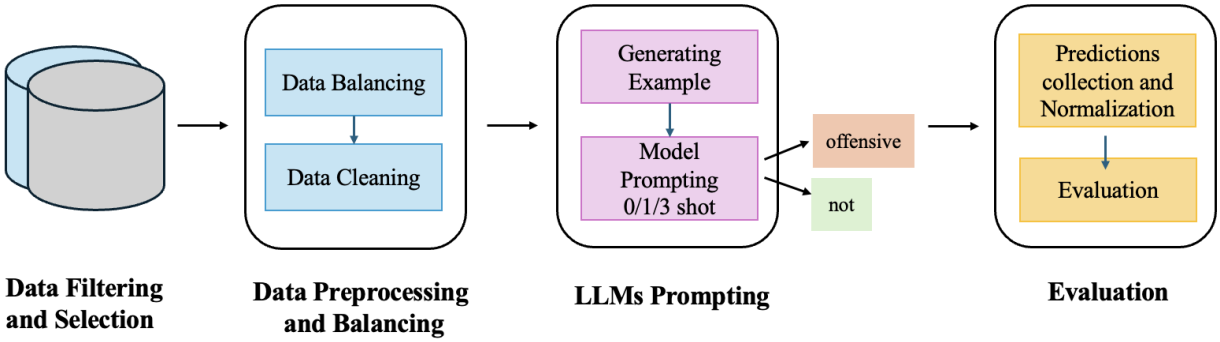

Fig. 1. Overview of the proposed methodology for Arabic hate speech classification using LLMs.

speech datasets, each annotated with binary labels. Each entry includes the tweet text, label, data source (e.g., Twitter), original dataset name, number of annotators, tweet ID, and the country of the post's author (inferred using user location data and geocoding APIs). Although the dataset covers a variety of Arabic dialects, such as Egyptian, Levantine, and Maghrebi, we filtered by dataset name to extract only the tweets originating from Saudi datasets. This decision was driven by the linguistic and cultural diversity across the Arab region. For example, Saudi Arabia alone has multiple distinct dialects, including Hijazi, Najdi, Janubi, and Hasawi [29]. Focusing on one regional context helps minimize dialectal variation and allows us to capture key sociolinguistic features specific to the Saudi context.

The filtering process resulted in a total of 3,178 tweets, with 2,328 labeled as not offensive and 850 as offensive. As is common in hate speech datasets, the distribution is imbalanced due to the relative infrequency of hateful content compared to neutral or non-offensive posts [28], [30]. To address this issue, and to understand evaluation bias while managing LLM cost constraints, we selected an equal number of instances from both classes. The final dataset used in our experiments consists of 1,600 tweets

Each tweet is associated with a human-annotated label based on its content (Offensive/Not Offensive). Prior to experimentation, we applied preprocessing to remove non-Arabic characters, mentions, hashtags, redundant whitespace, emoji-only entries, and tweets composed solely of punctuation or symbols. Such preprocessing is a common step in tweet classification studies, removing mentions and hashtags reduces noise and helps prevent the model from relying on user-specific or topic-specific cues that may not generalize well [31].

B. Models and Experiment Settings

To investigate the effectiveness of LLMs in detecting hate speech in Arabic tweets, we evaluate a range of state-of-the-art models. We utilized the following models: LLaMA3 `(llama3-70b)` [23], Gemma2 `(Gemma2-9b-it)` [24], GPT-4o [8], and ALLaM-7B [25]. Table I provides an overview of the selected models. These models were selected based on their popularity, open-access availability, and support for Arabic or multilingual processing. Each model was tested under zero-shot, one-shot, and few-shot prompts to assess how well they handle the task without task-specific fine-tuning.

TABLE I. SUMMARY OF THE LANGUAGE MODELS USED

Model	Parameters	#Tokens
llama3-70B	8 Billion	15T+
Gemma2-9B	9 Billion	8T
GPT-4o	∼1.8 Trillion	Not Disclosed
ALLaM-7B	7 Billion	500B

As a baseline, we used the state-of-the-art AraBERT[2] model. This model is based on XLM-RoBERTa [32] and was previously fine-tuned on Arabic toxic language datasets. We used the publicly available model without additional fine-tuning, to serve as an Arabic-specific baseline. Its role in this study is to provide a comparison point with traditional pretrained transformers adapted for Arabic.

All experiments were conducted in Python, utilizing libraries such as `transformers` for model interaction and `scikit-learn` for evaluation. Google Colab was used as the primary development environment. Each model was accessed via its respective API or hosted environment, using standardized prompts for consistency across experiments. Open-source models like Gemma2 and LLaMA3 were run through the Groq API [33]. GPT-4o, was accessed using the official OpenAI API. ALLaM, due to its computational demands, was executed locally using an NVIDIA T4 GPU and applied via Hugging Face's `AutoModelForCausalLM`. Finally, AraBERT was accessed via the Hugging Face `pipeline` interface.

C. Prompts Design

Prompt design is crucial when working with LLMs, as it directly influences how models interpret tasks and generate responses [34]. For hate speech classification, especially in Arabic, with its rich dialectal and cultural diversity, a clear and consistent prompt ensures reliable, comparable results across models. In this work, a standardized prompt was created carefully to ensure consistent evaluation across all models and experimental settings. The prompt clearly instructs the model to classify Arabic tweets as either offensive or not,

[2]https://huggingface.co/akhooli/xlm-r-large-arabic-toxic accessed 15 May 2025

while emphasizing concise response and preventing the models from generating explanations. The same prompt template was applied across all models and shots (0-, 1-, few-shot) to ensure consistent evaluation. The format of the prompts is illustrated in Table II.

Including one-shot and few-shot examples in the prompt is important because they provide concrete demonstrations of the classification task, which has been shown to significantly improve LLMs accuracy and reliability, especially for nuanced or low-resource tasks [35], [10]. Providing real, annotated examples helps the model understand the classification task and produce outputs that reflect human judgment. This is particularly valuable for Arabic hate speech detection, where context and subtle linguistic cues can vary widely across dialects. For the one-shot and three-shot settings, classification examples were randomly sampled from the dataset to minimize selection bias.

D. Evaluation Metrics

To assess the performance of the models, we used standard classification metrics including Precision, Recall, F1-score, and Accuracy. These metrics are defined as follows:

Precision measures how many of the predicted offensive tweets are actually offensive.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall measures how many of the actual offensive tweets were correctly predicted.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score is the harmonic mean of Precision and Recall, providing a balanced metric.

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Accuracy measures the overall proportion of correct predictions.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Here, TP (True Positives) refers to correctly predicted offensive tweets, FP (False Positives) are non-offensive tweets incorrectly predicted as offensive, FN (False Negatives) are offensive tweets missed by the model, and TN (True Negatives) are correctly predicted non-offensive tweets.

IV. Results

Each LLM was evaluated under three prompting configurations: zero-shot, one-shot, and few-shot. In the zero-shot setting, models were asked to classify tweets without any examples. The one-shot and few-shot settings included one and three labeled examples, respectively, embedded within the prompt. All models received the same classification prompt format to ensure consistency, and their predictions were compared against human-annotated labels.

Table III presents the performance of all evaluated models: AraBERT, LLaMA3, Gemma2, GPT-4o, and ALLaM on the Arabic hate speech classification task. We report Accuracy, Precision, Recall, and F1-score. All LLMs were able to outperform the baseline (AraBERT) across all metrics with F1-scores ranging from 0.75 to 0.86, compared to AraBERT's 0.71. GPT-4o consistently achieved the highest performance across all metrics and shot settings, with its F1-score improving from 0.83 (zero-shot) to 0.86 (three-shot). LLaMA3 and Gemma2 also demonstrated strong performance, showing consistent gains with the additional in-context examples. As the only Arabic-specific model in our evaluation, ALLaM started with a lower F1-score (0.75) in zero-shot, but improved significantly to 0.82 in the three-shot setup.

To better understand the level of agreement between the top-performing models, GPT-4o, LLaMA3, Gemma2, and ALLaM, we computed Fleiss' Kappa, a statistical measure of inter-rater reliability designed for evaluating consistency across multiple raters or classifiers [36]. A resulting score of 0.726 indicates substantial agreement, with values above 0.60 generally considered strong [37], suggesting that despite architectural and training differences, these LLMs produce largely consistent classification outcomes. As a result, the potential benefits of ensembling are reduced, since high agreement implies limited diversity in model outputs. Indeed, a majority-vote ensemble of the four models yielded performance nearly identical to GPT-4o alone (F1 = 0.86), reinforcing the conclusion that GPT-4o already captures most of the predictive power available across models. This finding aligns with recent studies, which report that ensemble methods offer the greatest benefit when base models are diverse in their predictions, and that high inter-model agreement can limit the effectiveness of majority-vote ensembles in NLP tasks [38], [39].

To assess the impact of class distribution on LLMs performance on the task at hand, we also evaluated all models on an imbalanced sample 1160:439, which reflects the original class distribution found in the dataset. This comparison provides insight into model robustness and real-world applicability. Table IV summarizes these results, with all metrics reported as macro-averages to ensure fair comparison across classes.

As expected, overall accuracy increased for most models compared to the balanced setting, since the majority class dominates the prediction task. However, macro F1-scores and recall values did not always improve, highlighting the challenges of maintaining balanced performance across all classes when the data is skewed. GPT-4o maintained the strongest results across all settings, achieving a macro F1-score of 0.84 in both the zero-shot and three-shot configurations. LLaMA3 and ALLaM also performed well in the three-shot setting, with F1-scores of 0.82 and 0.79, respectively. Gemma2's performance varied

TABLE II. PROMPT DESIGN EXAMPLES FOR 0-SHOT AND FEW-SHOT SETTINGS

0-shot Prompt	**Few-shot Prompt (e.g., 1-shot)**

Classify the input Arabic tweet as 'offensive' or 'not'.
The text will be delimited by triple backticks.
Answer only with 'offensive' or 'not'. Do not explain your answer!
``` tweet content ```
	Classify the input Arabic tweet as 'offensive' or 'not'.
	The text will be delimited by triple backticks.
	Answer only with 'offensive' or 'not'.
	Examples:
	``` Example tweet 1 ``` → offensive
	``` Example tweet 2 ``` → not
	Now classify:
	``` tweet content ```

TABLE III. PERFORMANCE OF LLMS ON THE (BALANCED) DATASET

Model		**Accuracy**	**Precession**	**Recall**	**F1 score**
AraBERT		0.71	0.72	0.71	0.71
LLaMA3	0-shot	0.79	0.80	0.79	0.79
	1-shot	0.82	0.83	0.82	0.82
	3-shot	0.83	0.84	0.83	0.83
Gemma2	0-shot	0.79	0.82	0.79	0.79
	1-shot	0.80	0.82	0.80	0.80
	3-shot	0.82	0.83	0.82	0.82
GPT-4o	0-shot	0.83	0.84	0.83	0.83
	1-shot	0.85	0.87	0.85	0.85
	3-shot	**0.86**	**0.87**	**0.86**	**0.86**
ALLaM	0-shot	0.76	0.79	0.76	0.75
	1-shot	0.78	0.82	0.78	0.78
	3-shot	0.82	0.82	0.81	0.81

more across shots, but still benefited from the addition of more in-context examples. It reached an F1-score of 0.79 in the three-shot setting despite lower scores at zero- and one-shot settings. AraBERT, serving as the baseline, achieved a macro F1-score of 0.70, which was outperformed by all LLMs.

TABLE IV. PERFORMANCE OF LLMS ON THE (IMBALANCED) DATASET. ALL METRICS ARE IN MACRO-AVERAGE

Model		**Accuracy**	**Precision**	**Recall**	**F1 score**
AraBERT		0.75	0.69	0.71	0.70
LLaMA3	0-shot	0.85	0.81	0.80	0.80
	1-shot	0.81	0.77	0.83	0.79
	3-shot	0.85	0.81	0.83	0.82
Gemma2	0-shot	0.74	0.74	0.80	0.72
	1-shot	0.71	0.73	0.79	0.70
	3-shot	0.81	0.77	0.83	0.79
GPT-4o	0-shot	**0.87**	**0.83**	0.85	**0.84**
	1-shot	0.84	0.80	**0.87**	0.82
	3-shot	0.86	0.83	0.85	**0.84**
ALLaM	0-shot	0.84	0.81	0.77	0.78
	1-shot	0.78	0.76	0.82	0.76
	3-shot	0.82	0.78	0.82	0.79

These findings confirm that LLMs are generally robust even under imbalanced conditions, particularly when few-shot prompts are provided. However, they also reinforce the importance of reporting macro-averaged metrics in hate speech classification tasks, as high accuracy alone may mask poor performance on the minority class (e.g. offensive content). While the overall performance of LLMs was strong, a closer examination of misclassifications provides further insight into model behavior, as discussed in the following section.

V. DISCUSSION

Our findings shows that LLMs are highly effective for Arabic hate speech detection, with GPT-4o achieving the highest F1-score (0.86) in the three-shot setting. This strong performance reflects GPT-4o's multilingual generalization abilities, as documented in prior work [27]. Additionally, open-source models like LLaMA3 and Gemma2 performed competitively, especially when guided with well-structured prompts, demonstrating that they are suitable for Arabic NLP tasks [40]. Moreover, the improvement in ALLaM's F1-score from 0.75 (zero-shot) to 0.82 (three-shot) aligns with patterns observed in other Arabic-centric models, such as JAIS [12], where the inclusion of contextual examples substantially narrows the gap between pretraining objectives and downstream task performance.

These findings suggest that prompt-based learning strategies are particularly effective for hate speech detection in Arabic, a domain where linguistic nuance and cultural context play a significant role in defining classification boundaries [15]. The observed effectiveness of in-context examples supports theoretical frameworks proposing that LLMs are capable of capturing implicit sociolinguistic cues, such as sarcasm and dialectal variation, when provided with representative examples [41], [42]. This is especially critical in Arabic, where hate speech often appears through indirect references and dialect-specific expressions that pose challenges to static lexicon-based approaches [15]. However, a deeper analysis of model errors is essential to understand the nature and sources of misclassifications, and to guide future improvements, especially in the context of Saudi Arabia.

To gain deeper insights into model behavior, we conducted a comprehensive error analysis combining confusion matrices and manual inspection of misclassified instances. Fig. 2 displays the confusion matrices for all models under the three-shot setting using the balanced dataset, while Fig. 3 illustrates performance on the imbalanced dataset. These visualizations highlight distinct error patterns shaped by class distribution and reveal how each model handles nuanced classification scenarios.

The first category of error involves class-specific misclassification patterns. The confusion matrices reveal that the distribution of errors differs significantly between the balanced and imbalanced datasets. In the balanced setting, models tend to generate more false positives than false negatives. For instance, GPT-4o misclassified 17.5% of non-offensive tweets as offensive (140 out of 800), whereas its false negative rate was 9.9% (79 out of 800 offensive tweets misclassified as non-offensive). However, this trend is reversed in the imbalanced

setting: GPT-4o's false negative rate increases to 17.1% (75 out of 439), while false positives drop to 12.1% (141 out of 1161). This shift illustrates the model's tendency to favor the majority class in skewed distributions, a common issue in imbalanced classification tasks [41], [17].

The second category of error is Model-Specific Vulnerabilities. Gemma2 showed the most notable sensitivity to data distribution, with a consistently high false positive rate, 27.3% in the balanced setting (218 out of 800) and 21.1% in the unbalanced setting (245 out of 1161). This pattern suggests a persistent tendency to misclassify non-offensive content as offensive, possibly due to limitations in handling subtle contextual cues. In contrast, ALLaM demonstrated more stable behavior across distributions, with false positive rates of 23.8% in the balanced setting (190 out of 800) and 18.1% in the unbalanced setting (210 out of 1161). This relatively moderate error fluctuations indicates stronger fine-tuning for Arabic hate speech detection despite having slightly lower overall accuracy than GPT-4o or LLaMA3.

The Third category of error stems from linguistic and contextual challenges. To further investigate these challenges, we conducted a manual error analysis on a stratified sample of 60 misclassified tweets, with 15 instances drawn from each of the three-shot settings of the top-performing models, i.e. GPT-4o, LLaMA3, Gemma2, and ALLaM. Each misclassified tweet was annotated with a single dominant error category based on linguistic, semantic, or contextual cues that likely contributed to the misclassification. Fig. 4 illustrates the distribution of error categories among the misclassified sample.

The most common error category involved tweets expressing implicit hate and indirect language, which accounted for 30% of the annotated sample, including 7% explicitly marked as sarcasm. These tweets often relied on cultural references, regional superiority, or subtle phrasing rather than directly offensive language. For example, the statement أنا من أهل [..] لم أسمع بهذه العادة من قبل! من أين تأتون بها؟ ("I'm from the [..] region, and I've never heard of this custom before. Where are you bringing it from?") implicitly conveys cultural exclusion and superiority without directly attacking a group. By framing the custom as unfamiliar or foreign, the speaker questions its authenticity and implicitly discriminates against those who practice it. Such instances of indirect hate pose a challenge for large language models, which often rely on explicit lexical indicators. Detecting these subtle nuances requires deeper context awareness and cultural understanding, areas where current models still face limitations.

Another major source of error is dialect ambiguity observed in 20% of the sample. Tweets written in regional dialects such as Najdi or Hijazi, or those incorporating local slang, were frequently misclassified. This is likely due to the under-representation of these dialects in most LLM pretraining corpora. Dialectal differences in vocabulary, syntax, and idiomatic expressions can significantly alter the perceived offensiveness of a statement, particularly when models lack targeted exposure to these variations. Additionally, some expressions observed in the dataset reflected linguistic influence from neighboring regions, such as Gulf and Egyptian dialects, which are naturally present in the Saudi linguistic landscape. This cross-regional blending adds further complexity to model interpretation, especially when the cues for offensiveness are subtle or culturally specific.

Misclassifications also occurred in 17% of tweets that mentioned named individuals or groups without providing sufficient contextual cues. This type of named entity confusion arises when a model lacks the external or background knowledge needed to determine whether a reference is offensive or neutral. In many cases, tweets rely on insider knowledge, local discourse, or references to public figures, making it difficult for models, particularly in zero- or few-shot settings, to correctly interpret intent. For instance, the tweet "بكاء الحقوقيات طرب" ("The crying of feminists is like music") may appear neutral or poetic. However, depending on the social context, it may carry sarcasm or targeted implications. Another tweet, "رواه البخاري ومسلم", references a well-known hadith phrase, but when used outside religious discourse, it can be employed mockingly or to dismiss opposing views. Therefore without the relevant cultural grounding, models may either over- or under-classify such tweets, leading to errors. This highlights the limitation of LLMs in performing context-aware reasoning and culturally informed classification in low-context tasks.

Another challenge observed in 10% of the sample was topic bias. This occurs when the model misclassifies tweets as offensive or non-offensive based primarily on the subject matter or specific keywords, rather than the actual tone or context. Tweets discussing sensitive or controversial topics, such as politics, religion, or social issues, may be flagged as offensive even when they are neutral or factual. For instance, terms like اليهود, and إدمان ("jewish, and addiction") can trigger false positives due to their frequent association with offensive content during training. A relevant example is: "مع ايقاف اللعبة وبشده صرنا مدمنين عليها بشكل" ("I strongly support banning this game … we've become terribly addicted to it."), which is factually neutral but was incorrectly flagged as offensive. However, tweets containing harmful language on less controversial topics may escape detection if the model has implicitly learned to associate those subjects with with non-offensive speech. This bias is due to the model overfitting to patterns in the training data, where certain topics are far less represented compared to offensive speech.

The final category of errors relates to annotation ambiguity, accounting for 10% of the sample. In these cases, the ground truth labels themselves appeared questionable or inconsistent. This reflects the inherent subjectivity involved in labeling hate or offensive speech, especially in borderline cases or when annotators apply differing cultural or personal interpretations. Such ambiguity introduces label noise, which not only affects model training but also complicates evaluation by obscuring what constitutes a correct prediction. For example, the tweet "يا[–] جيزان عسير نجران أراضي [–]" ("[Group] [Region names] belong to [Country]") was annotated as not offensive. However, it arguably conveys an implicit denial of national identity, raising concerns about annotation consistency. This highlights how politically sensitive claims, when expressed implicitly, may be overlooked by annotators, reinforcing the importance of clear annotation criteria and sociopolitical awareness in training data curation.

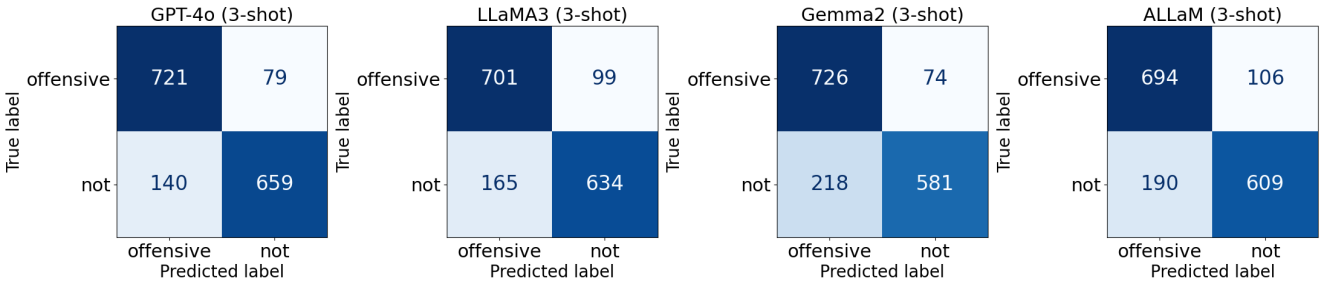

Fig. 2. Confusion matrices of LLMs under three-shot setting on the balanced dataset.

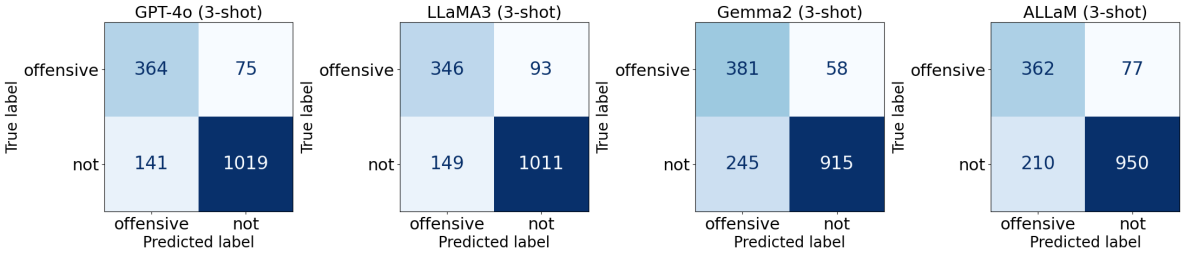

Fig. 3. Confusion matrices of LLMs under three-shot setting on the unbalanced dataset.

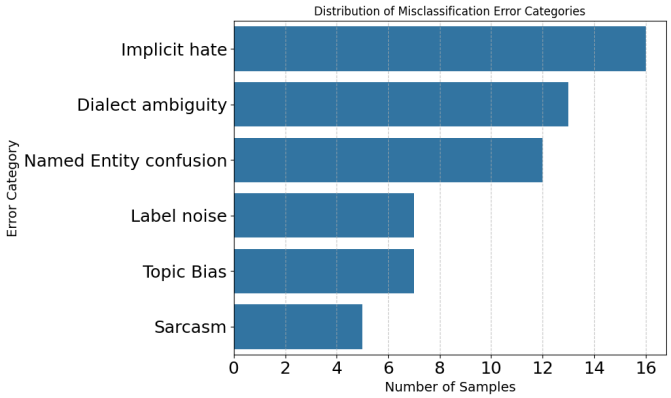

Fig. 4. The distribution of misclassifications error categories.

Our analysis reveals several persistent challenges that extend beyond technical limitations. These challenges are deeply rooted in the sociolinguistic complexity of Arabic, as well as in the subjective nature of annotating hate speech. Addressing such issues requires both linguistic sensitivity and culturally aware annotation strategies.

VI. Conclusion

This study examined the capabilities of LLMs in classifying Arabic hate speech, with a particular focus on prompt-based learning and sociolinguistic differences within Saudi dialects. The evaluation was performed under different prompting settings, including zero-shot, one-shot, and three-shot. While GPT-4o outperformed all models, open-source models such as LLaMA3, Gemma2, and ALLaM also demonstrated competitive performance. Moreover, all models benefited from the support of relevant contextual examples. A key finding is that well-structured prompts significantly enhance model performance, further supporting the potential of prompt engineering in low-resource language settings.

Our in-depth error analysis revealed that misclassifications stem from a mix of technical and sociolinguistic challenges. Common sources of error included implicit hate, dialectal variation, named entity confusion, topic bias, and annotation ambiguity. These findings highlight the need for culturally informed evaluation, richer dialectal representation, and improved annotation guidelines in Arabic NLP tasks. While the evaluation was conducted on a representative subset of the dataset, this approach aligns with established practices in LLM assessment and offers reliable insights into model behavior across larger corpora.

Beyond benchmarking, improved hate speech detection in Arabic dialects can support safer online spaces, strengthen content moderation systems, and inform more nuanced policy-making in multilingual contexts. Addressing annotation ambiguity also calls for culturally grounded guidelines and tools that better support annotators. Future research should explore hybrid modeling approaches, culturally enriched training data, dialect-specific fine-tuning, and more representative few-shot examples to further enhance prompt-based performance.

References

[1] A. Alhazmi, R. Mahmud, N. Idris, M. E. M. Abo, and C. Eke, "A systematic literature review of hate speech identification on arabic twitter data: research challenges and future directions," *PeerJ Computer Science*, vol. 10, p. e1966, 2024.

[2] A. Omar, T. M. Mahmoud, and T. Abd-El-Hafeez, "Comparative performance of machine learning and deep learning algorithms for arabic hate speech detection in osns," in *Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020)*. Springer, 2020, pp. 247–257.

[3] M. Hietanen and J. Eddebo, "Towards a definition of hate speech—with a focus on online contexts," *Journal of Communication Inquiry*, vol. 47, no. 4, pp. 440–458, 2023.

[4] UNESCO, "World arabic language day," n.d., accessed: 2025-06-10. [Online]. Available: https://www.unesco.org/ar/world-arabic-language-day

[5] A. Alakrot, L. Murray, and N. S. Nikolov, "Dataset construction for the detection of anti-social behaviour in online communication in arabic," in *Procedia Computer Science*, vol. 142. Elsevier, 2018, pp. 174–181.

[6] A. Charfi, M. Besghaier, R. Akasheh, A. Atalla, and W. Zaghouani, "Hate speech detection with adhar: a multi-dialectal hate speech corpus in arabic," vol. 7. Frontiers Media SA, 2024, p. 1391472.

[7] T. Alhindi, P. Ghaffari, and W. Magdy, "Fine-tuned hate speech detection in arabic social media," *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, vol. 20, no. 2, pp. 1–19, 2021.

[8] OpenAI, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[9] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[10] M. Zouidine and M. Khalil, "Large language models for arabic sentiment analysis and machine translation," *Engineering, Technology & Applied Science Research*, vol. 15, no. 2, pp. 20 737–20 742, 2025.

[11] A. Atef, F. Seddik, and A. Elbedewy, "Ags: Arabic gpt summarization corpus," in *2023 International Conference on Electrical, Communication and Computer Engineering (ICECCE)*. IEEE, 2023, pp. 1–8.

[12] A. A. Al Wazrah, A. Altamimi, H. Aljasim, W. Alshammari, R. Al-Matham, O. Elnashar, M. Amin, and A. AlOsaimy, "Evaluation of large language models on arabic punctuation prediction," in *Proceedings of the 1st Workshop on NLP for Languages Using Arabic Script*, 2025, pp. 144–154.

[13] M. Mozafari, R. Farahbakhsh, and N. Crespi, "A bert-based transfer learning approach for hate speech detection in online social media," in *International conference on complex networks and their applications*. Springer, 2019, pp. 928–940.

[14] W. Antoun, F. Baly, and H. Hajj, "Arabert: Transformer-based model for arabic language understanding," *arXiv preprint arXiv:2003.00104*, 2020.

[15] H. Mubarak, S. Hassan, and S. A. Chowdhury, "Emojis as anchors to detect arabic offensive language and hate speech," *Natural Language Engineering*, vol. 29, no. 6, pp. 1436–1457, 2023.

[16] M. Tonneau, D. Liu, S. Fraiberger, R. Schroeder, S. Hale, and P. Röttger, "From languages to geographies: Towards evaluating cultural bias in hate speech datasets," in *Proceedings of the 8th Workshop on Online Abuse and Harms (WOAH 2024)*. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 283–311. [Online]. Available: https://aclanthology.org/2024.woah-1.23

[17] A. Ahmad, M. Azzeh, E. Alnagi, Q. Abu Al-Haija, D. Halabi, A. Aref, and Y. AbuHour, "Hate speech detection in the arabic language: corpus design, construction, and evaluation," *Frontiers in Artificial Intelligence*, vol. 7, p. 1345445, 2024.

[18] R. Alshalan and H. Al-Khalifa, "A deep learning approach for automatic hate speech detection in the saudi twittersphere," *Applied Sciences*, vol. 10, no. 23, p. 8614, 2020.

[19] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *ACM transactions on intelligent systems and technology*, vol. 15, no. 3, pp. 1–45, 2024.

[20] M. Abdul-Mageed, A. Elmadany, and E. M. B. Nagoudi, "Arbert & marbert: Deep bidirectional transformers for arabic," *arXiv preprint arXiv:2101.01785*, 2020.

[21] N. Wies, Y. Levine, and A. Shashua, "The learnability of in-context learning," *Advances in Neural Information Processing Systems*, vol. 36, pp. 36 637–36 651, 2023.

[22] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[23] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[24] G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé *et al.*, "Gemma 2: Improving open language models at a practical size," *arXiv preprint arXiv:2408.00118*, 2024.

[25] M. S. Bari, Y. Alnumay, N. A. Alzahrani, N. M. Alotaibi, H. A. Alyahya, S. AlRashed, F. A. Mirza, S. Z. Alsubaie, H. A. Alahmed, G. Alabduljabbar *et al.*, "Allam: Large language models for arabic and english," *arXiv preprint arXiv:2407.15390*, 2024.

[26] P. Dadure, A. Dixit, K. Tewatia, N. Paliwal, and A. Malla, "Sentiment analysis of arabic tweets using large language models," in *Proceedings of the 1st Workshop on NLP for Languages Using Arabic Script*, 2025, pp. 88–94.

[27] S. Khaled, E. H. Mohamed, and W. Medhat, "Evaluating large language models for arabic sentiment analysis: A comparative study using retrieval-augmented generation," *Procedia Computer Science*, vol. 244, pp. 363–370, 2024.

[28] S. Alghamdi, Y. Benkhedda, B. Alharbi, and R. T. Batista-Navarro, "Aratar: A corpus to support the fine-grained detection of hate speech targets in the arabic language," in *Proceedings of the 6th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT) with Shared Tasks on Arabic LLMs Hallucination and Dialect to MSA Machine Translation@ LREC-COLING 2024*, 2024, pp. 1–12.

[29] T. Alqurashi, "Applying a character-level model to a short arabic dialect sentence: a saudi dialect as a case study," *Applied Sciences*, vol. 12, no. 23, p. 12435, 2022.

[30] K. Madukwe, X. Gao, and B. Xue, "In data we trust: A critical analysis of hate speech detection datasets," in *Proceedings of the fourth workshop on online abuse and harms*, 2020, pp. 150–161.

[31] D. Ramachandran and R. Parvathi, "Analysis of twitter specific preprocessing technique for tweets," *Procedia Computer Science*, vol. 165, pp. 245–251, 2019.

[32] S. Ruder, A. Søgaard, and I. Vulić, "Unsupervised cross-lingual representation learning," in *Proceedings of the 57th annual meeting of the association for computational linguistics: Tutorial abstracts*, 2019, pp. 31–38.

[33] G. Inc. (2024) Groq api documentation. Accessed May 12, 2025. [Online]. Available: https://docs.groq.com

[34] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, and D. C. Schmidt, "A prompt pattern catalog to enhance prompt engineering with chatgpt," *arXiv preprint arXiv:2302.11382*, 2023.

[35] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[36] K. L. Gwet, "Large-sample variance of fleiss generalized kappa," *Educational and Psychological Measurement*, vol. 81, no. 4, pp. 781–790, 2021.

[37] F. Moons and E. Vandervieren, "Measuring agreement among several raters classifying subjects into one-or-more (hierarchical) nominal categories. a generalisation of fleiss' kappa," *arXiv preprint arXiv:2303.12502*, 2023.

[38] H. Mubarak, K. Darwish, W. Magdy, T. Elsayed, and H. Al-Khalifa, "Overview of osact4 arabic offensive language detection shared task," in *Proceedings of the 4th Workshop on open-source arabic corpora and processing tools, with a shared task on offensive language detection*, 2020, pp. 48–52.

[39] K. E. Daouadi, Y. Boualleg, and K. E. Haouaouchi, "Ensemble of pre-trained language models and data augmentation for hate speech detection from arabic tweets," *arXiv preprint arXiv:2407.02448*, 2024.

[40] M. T. I. Khondaker, N. Naeem, F. Khan, A. Elmadany, and M. Abdul-Mageed, "Benchmarking llama-3 on arabic language generation tasks," in *Proceedings of The Second Arabic Natural Language Processing Conference*, 2024, pp. 283–297.

[41] K. E. Daouadi, Y. Boualleg, and O. Guehairia, "Comparing pre-trained language model for arabic hate speech detection," *Computación y Sistemas*, vol. 28, no. 2, pp. 681–693, 2024.

[42] J. M. Pérez, P. Miguel, and V. Cotik, "Exploring large language models for hate speech detection in rioplatense spanish," *arXiv preprint arXiv:2410.12174*, 2024.