GraphRAG-based NLP at Risk: Graphemic Dot-Level Adversarial Attack on Arabic Sentiment and LLM Retrieval-Augmented Models

Abdullah Melhem[1], Ahmed Aleroud[1], and Craig Douglas Albert[2]

[1] *School of Computer and Cyber Sciences, Augusta University, Augusta, GA, USA*
abanimelhem@augusta.edu, aaleroud@augusta.edu
[2] *National Defense Studies, Augusta University, Augusta, GA, USA*
calbert@augusta.edu

Abstract While research on adversarial attacks has advanced significantly, most studies on Natural Language Processing (NLP) have predominantly focused on English, leaving the vulnerabilities of models trained on other languages largely unexplored. These attacks pose a direct challenge to the reliability of AI systems used to interpret Arabic-language content on social media platforms, where sentiment analysis and content moderation tools are routinely deployed. This study introduces a novel graphemic dot-level adversarial attack specifically designed to target large language models (LLMs) trained on Arabic text. Unlike traditional adversarial attacks, our method manipulates dots within Arabic characters, leveraging common spelling errors made by non-native Arabic speakers to create imperceptible, deceptive, and highly effective adversarial examples. These modifications, though minimal, significantly degrade the performance of widely used Arabic text Machine Learning (ML) classifiers such as sentiment analysis models, and the responses of LLMs such as GPT-4o-mini used in LLM-based Retrieval-Augmented Generation (RAG) systems. Our experiments reveal that even advanced LLM-driven retrieval models, which rely on graph knowledge to enhance response accuracy, remain highly susceptible to our fine-grained perturbations. Our results, using Telegram data sets, demonstrate that offensive AI is effective in NLP models for low-resource languages such as Arabic and emphasize the need for defense mechanisms to mitigate the impact of such adversarial manipulations. [3] [4]

Keywords: Adversarial attacks, Large language models, RAG, Sentiment analysis, AI

1 Introduction

The recent advancements in offensive adversarial attacks have gained significant attention in the domain of NLP due to their ability to manipulate ML/AI predictions by introducing subtle modifications to input data [3]. While extensive research has been

[3] The code and data are publicly available in the authors' GitHub repository

[4] This material is based upon research supported by the U.S. Office of Naval Research under award number N000142212549.

conducted on adversarial attacks in English [13], limited efforts have explored the vulnerabilities of DNNs and AI models trained on low-resource languages such as Arabic [1]. Given the structural complexity of the Arabic language [5], which includes diacritical marks, dot-based character variations, and morphological richness, traditional adversarial attack strategies may not be directly applicable in this context. A key challenge lies in crafting perturbations that are both stealthy and effective. [16]. This gap in research highlights the need for stealthy adversarial techniques that exploit language-specific characteristics to evaluate the robustness of Arabic NLP models. Furthermore, while social media users attempt to bypass content filtering algorithms by manually applying adversarial perturbations to text, the use of offensive AI techniques poses an even greater threat to content moderation. These advanced methods can weaken the accuracy of models designed to detect disinformation, propaganda, hate speech, and anti-ethnic content on social media, making them more vulnerable to manipulation.

In this paper, we introduce a novel graphemic manipulation stealthy attack targeting Arabic sentiment classification models, LLMs, and LLM-driven GraphRAG systems. Using gradient-based optimization function, our blackbox attack systematically manipulates Arabic text at the character level by flipping dot placements, mimicking common spelling mistakes made by non-native Arabic learners. This form of attack is particularly effective because many Arabic letters are differentiated solely by the presence or absence of diacritics, leading to significant changes in meaning with minimal textual perturbation. We introduce a multi-level attack that incrementally applies perturbations to maximize sentiment shifts. We demonstrate how minor modifications can degrade the performance of LLMs and mislead LLM-driven retrieval models. To assess the effectiveness of our attack, we conduct experiments on various Arabic text datasets that we collected from Telegram and other existing datasets, as detailed in section 3.1. The results reveal that even powerful models trained on large-scale Arabic corpora remain highly susceptible to these offensive perturbations, affecting both sentiment predictions and the correctness of retrieved knowledge. Not only does it disrupt sentiment classification, but it also introduces inaccurate AI generative responses, making it particularly dangerous for real-world NLP social media applications.

The rest of this paper is structured as follows: Section 2 presents a review of related adversarial attack techniques in NLP, focusing on methods applied to Arabic text. Section 3 details our proposed methodology, including the adversarial attack framework. Section 4 provides a comprehensive analysis of the results obtained from various sentiment analysis models, and demonstrates the impact of our attack on GraphRAG-based AI retrieval systems. Finally, Section 5 concludes the paper and discusses potential directions for future research.

2 Related work

In this section, we review existing adversarial attack techniques in NLP, with a particular focus on offensive methods applied to text. We discuss character-level and word-level offensive attacks, highlight their effectiveness, and limitations in assessing the robustness of deep learning models. Character-level adversarial attacks expose vulnerabilities in NLP models by subtly altering text through character insertion, deletion, or substitution. These changes often go unnoticed by humans but can mislead models significantly. HotFlip attacks flip characters using gradient information [8], while other approaches have demonstrated that random perturbations can degrade the accuracy of sentiment models, dropping it from 90% to as low as 45.8% [14]. Liu et al. highlighted the challenges in developing effective defenses, particularly due to out-of-vocabulary (OOV) issues and discrepancies between training and inference distributions [12].

TextAttack was introduced by Morris et al. as a framework for data augmentation and adversarial training. It modularizes attack construction via goal functions, constraints, transformations, and search methods, supporting models such as BERT and GLUE tasks [13]. Adversarial text attacks have been explored in other languages. TextGuise [7] attacks English models using synonym substitution, emojis, and dictionary edits, achieving 80% success with minimal disruption. Argot [17] targets Chinese models using pinyin-based changes, glyph edits, and character shuffling, reaching over 97% success. Recent research has explored various adversarial attack strategies targeting Arabic text classification models, each focusing on different linguistic levels. Alshemali and Kalita [3] introduced a character-level attack that manipulates input text by introducing spelling errors and replacing characters with visually similar Arabic letters. Similarly, Radman and Duwairi [15] showed that sentiment analysis models are susceptible to synonym replacement attacks guided by gradient information. They compared different adversarial training strategies, including mixing adversarial examples with clean data and applying weight perturbations. Their findings indicate that incorporating adversarial examples enhances model robustness without compromising accuracy.

3 Methodology

This section outlines the motivation and methodology behind our proposed adversarial attack. Prior studies on Arabic text processing highlight frequent spelling errors by non-native learners, often caused by phonetic similarities, visual letter resemblance, and diacritic confusion [4]. These consistent substitution patterns affect text comprehension and NLP performance. Building on this, we explore how such common errors can be leveraged to deceive deep learning sentiment analysis models. This technique takes advantage of the visual similarities between Arabic letters. For example, the character ت can be replaced with ث, ب, or ن, all of which share dot-based variations, while still producing readable content. Similarly, ب can be confused with ت, ث, or ن, and ث may be substituted with ت, ب, or ن. The letter ن itself is often interchanged with ب, ت, or ث. These substitutions target phonetic overlaps such as 't' in "top", 'b' in "bat", 'th' in "think", and 'n' in "no". The character ي may be substituted with ى, mimicking the sounds of 'y' in "yes" and 'aa' as in "bazaar", respectively. Additionally, ف and ق are often interchanged due to similar structure, representing the sounds 'f' as in "fun" and the deeper 'q' sound. In another example, the rolled ر can be confused with ز, which corresponds to 'z' as in "zebra". Our method differs from the approach in [3] in both the scoring strategy and the type of substitutions used. We substitute characters by modifying only the placement of dots, resulting in subtle changes, whereas their substitutions involve visually distinct characters (e.g., ـک and ل) that are easily noticeable to human readers.

While their method employs the Replace-1 Scoring Function (R1S), which identifies important tokens by replacing characters with out-of-vocabulary variants, our approach determines token importance by iteratively removing each word from the input and evaluating the change in sentiment using the pre-trained CamelBERT-Da-Sentiment model [11]. Formally, given a sentence $x = [w_1, w_2, \ldots, w_{i-1}, w_i, \ldots, w_n]$, the R1S function is different compared to ours as follows: R1S replaces the token, $x = [w_1, w_2, \ldots, w_{i-1}, w_i', \ldots, w_n]$; ours deletes it, $x = [w_1, w_2, \ldots, w_{i-1}, w_{i+1}, \ldots, w_n]$.

We argue that removal yields a more accurate importance signal than replacement, particularly for models with memory mechanisms like BiLSTM, where introducing unnatural tokens can disrupt contextual understanding. The word that causes the largest sentiment shift upon removal is identified as the most important.

3.1 Datasets

In this study, we collected several Telegram datasets to measure the impact of the attack on social platforms that support instant messaging. We also utilized other datasets from diverse domains to assess the performance of our adversarial attack. These datasets include both social media data and domain-specific reviews. Table 1 presents a summary of the datasets used. We collected Telegram messages from several public groups over different time intervals, spanning various topics such as politics, finance, political conflict, and technology. The data collection period varied for each group, with the longest spanning from December 2019 to September 2024. The total number of messages collected across all Telegram groups is approximately 1.05M, contributed by 24,920 unique users who posted at least twice. The datasets cover diverse domains, including political discussions, financial markets, technological advancements, and medical content. The detailed dataset specifications, including group size and domain, are outlined in Table 1. Additionally, we use the Hotel Arabic Reviews Dataset (HARD), which is a balanced dataset containing 93K hotel reviews, primarily written in Modern Standard Arabic (MSA) and collected from Booking.com [10]. We also utilized a sentiment analysis dataset for Arabic social media posts (MSDA), comprising 50K posts predominantly in Arabic, sourced from the X (formerly Twitter) platform [6].

Table 1. Dataset Information

Data Source	Domain	Size	Source
AlarabyTelevision	Political	41K	Telegram
BinanceArabic	Financial	206K	Telegram
Ehabtvv	Technology	191K	Telegram
Gazaalannet	Conflicts	113K	Telegram
JeninBoss	Conflicts	335K	Telegram
Kfoovip	General	145K	Telegram
Medicinal4	Medical	17K	Telegram
HARD	Hotel Reviews	93K	Booking.com
MSDA	Social Media Posts	50K	X platform

3.2 Threat Model

Under this attack, the adversaries operate in a black-box setting, meaning they do not have access to model parameters, gradients, or training data but can query the model and observe output probabilities $P(C(x) = y)$. The attacker modifies text at the character level, specifically targeting letters that differ only in diacritical dots. The attack function $g : X \to X_{\text{adv}}$ selects a vulnerable character c_i and replaces it with a visually similar character from the set M, ensuring that the total number of modified characters remains within a predefined budget $\sum_{i=1}^{n} \mathbb{I}(c_i \neq c_i^{\text{adv}}) \leq \epsilon$, where $\mathbb{I}(\cdot)$ is an indicator function.

The attack impacts robustness, as the classifier is easily misled by minor perturbations satisfying $||x - x_{\text{adv}}|| \leq \epsilon$ but resulting in $C(x_{\text{adv}}) \neq C(x)$. The stealthiness of the attack is achieved by perturbations that remain minimally perceptible to human readers since the edit distance $d(x, x_{\text{adv}})$ is minimized.

3.3 Adversarial example generation

The adversarial text generation process aims to create perturbed examples from a dataset $D = (X, Y)$, where each input-label pair $(x_i, y_i) \in \{(x_1, y_1), \ldots, (x_n, y_n)\}$ is classified by a black-box classifier $C : X \to Y$. In this setting, the classifier returns output labels Y and confidence scores P without exposing internal details such as parameters, gradients, or architecture. Each input $x \in X$, composed of W words $x = [w_1, w_2, \ldots, w_W]$ with label y, is used to generate adversarial examples X_{ADV} that change the model's prediction, i.e., $C(X_{\text{ADV}}) \neq C(X)$, while preserving grammatical correctness and semantic similarity to the original input.

We employ a perturbation strategy that identifies the most influential words in a given text using a gradient-based function. Specifically, we define a sentiment classification model $C : X \to Y$, where X represents an input text consisting of a sequence of words $X = [w_1, w_2, \ldots, w_n]$, and Y denotes the predicted sentiment label. The influence of each word w_i on the classification outcome is computed using an attention-based method, where the importance score is defined as $I(w_i) = \frac{\partial P(Y|X)}{\partial w_i}$, and the most influential word is identified as $w^* = \arg\max_{w_i \in X} I(w_i)$. Once the most impactful word is selected, adversarial perturbations are introduced by applying a substitution function $g : W \to W'$, which maps w^* to an adversarially modified word w^*_{adv}, ensuring that $w^*_{\text{adv}} \in S(w^*) = \{s_1, s_2, \ldots, s_k\}$, where $S(w^*)$ is a predefined set of semantically similar words, typos, or common spelling mistakes. The resulting adversarial text is then formulated as $X_{\text{adv}} = [w_1, \ldots, w^*_{\text{adv}}, \ldots, w_n]$, which aims to mislead the classifier while maintaining grammatical coherence. The attack is executed in a multi-level manner, where Level 1 (L1) replaces the most influential word w^*, Level 2 (L2) applies perturbations to the second most influential word $w^{**} = \arg\max_{w_i \in X \setminus \{w^*\}} I(w_i)$, and Level 3 (L3) modifies multiple characters within these words to maximize the sentiment shift. Compared to [3], where both attack levels (Ar-Flip and Ar-Flip2) are applied to the same token, potentially altering it completely, we distribute our attack across the first and second most important tokens to preserve semantic structure.

To execute the attack, we start by iterating through each word in the sentence, temporarily removing one word at a time. After each removal, we evaluate the sentiment score using the pre-trained CamelBERT-Da-Sentiment model [11]. The word that causes

Algorithm 1 Multi-Level Adversarial Dot-Based Attack

```
Require: Sentence $T$, substitution groups $S$
Compute sentiment scores of $T$: $(pos, neg, neu) \leftarrow$ compute_sent($T$)
$W \leftarrow$ Tokenize $T$ into words
$MIW \leftarrow$ Find the most important vulnerable word using max sentiment drift
$A \leftarrow$ Generate adversarial examples for $MIW$ using $S$
$T_1 \leftarrow$ Apply Attack Level 1: Replace $MIW$ in $T$ with most effective $A$
Compute sentiment change $(\Delta pos_1, \Delta neg_1, \Delta neu_1)$
$T_2 \leftarrow$ Apply Attack Level 2: Optimize perturbation for max $\Delta$
Compute sentiment change $(\Delta pos_2, \Delta neg_2, \Delta neu_2)$
$T_3 \leftarrow$ Apply Attack Level 3: Multi-word perturbation (if applicable)
return $T'$ with highest $\Delta$
```

the largest sentiment shift upon removal is identified as the most important word in the sentence (MIW) as detailed in algorithm 1. Once the keyword is determined, we analyze its individual characters to find the one that contributes the most to sentiment shift. The attack is deployed against two critical NLP components: (1) pre-trained sentiment models such as CamelBERT-Da classifiers and (2) GraphRAG-based retrieval systems, measuring their robustness against adversarial perturbations through systematic stress-testing.

3.4 Attacking Fine-tuned GraphRAG

We evaluated the response differences between two fine-tuned versions of the GraphRAG model [9], one trained on adversarial data and the other on clean data. GraphRAG is an LLM-powered, graph-enhanced knowledge retrieval system that improves contextual understanding by using knowledge graphs. In our setup, GPT-4o-mini, instantiated via an OpenAI model object, was used as the LLM within the GraphRAG framework. The evaluation involved fine-tuning GraphRAG twice—once with adversarially augmented data and once with clean data—then testing both versions using the following set of questions. These queries focused on sentiment analysis, community sentiment, sentiment propagation, and user-level sentiment.

Q1: Percentage distribution of messages classified as positive, negative, and neutral? *(Sentiment Analysis Overview)*
Q2: What is the ratio of positive to negative messages? *(Sentiment Analysis Overview)*
Q3: Which community has the largest proportion of negative sentiment messages? *(Community Sentiment)*
Q4: What is the total number of messages classified as positive, negative, and neutral? *(Community Sentiment)*
Q5: Which users are most influential in spreading positive or negative sentiment? *(Sentiment Propagation)*

The questions presented to both models aimed to assess their accuracy in reflecting sentiment distribution, community sentiment, sentiment propagation, and key dataset topics. The objective was to compare the responses of the adversarially fine-tuned model

Table 2. Sentiment predictions under our Dot-Level attack and the Ar-Flip2 method, demonstrating differences in effectiveness and subtlety between the two strategies.

Version	Arabic Sentence	Pos	Neg	Neu
Original	بذلت مجهودا جبارا في محاولة قراءة الكتاب لكنني فشلت القصة غريبة جداً لدرجة أنني توقفت ولم أستطع إكمال القراءة *I put tremendous effort into trying to read the book, but I failed. The story is so strange that I stopped and couldn't continue reading.*	**0.927**	0.27	0.0455
Dot-L1A	بذلت مجهودا جبارا في محاولة قراءة الكتاب لكنني قشلت القصة غريبة جداً لدرجة أنني توقفت ولم أستطع إكمال القراءة *I put tremendous effort into trying to read the book, but I qashalt. The story was so strange that I stopped and couldn't continue reading.*	**0.691**	0.201	0.107
Dot-L2A	بذلت مجهودا جنارا في محاولة قراءة الكتاب لكنني قشلت القصة غريبة جداً لدرجة أنني توقفت ولم أستطع إكمال القراءة *I put janaran effort into trying to read the book, but I qashalt. The story was so strange that I stopped and couldn't continue reading.*	0.305	**0.582**	0.112
Ar-Flip2	بذلت مجهودا جبارا في محاولة قراءة الكتاب لكنني فشلت القصة غزينه جداً لدرجة أنني توقفت ولم أستطع إكمال القراءة *I put tremendous effort into trying to read the book, but I failed. The story is so ghazina that I stopped and couldn't continue reading.*	**0.959**	0.007	0.032

Table 3. Accuracy of models under different attack levels (L1–L3).

Dataset	IMAMU			Fine-Tuned IMAMU			CamelBERT		
	L1	L2	L3	L1	L2	L3	L1	L2	L3
AlarabyTelevision	0.80	0.75	0.73	0.81	0.77	0.71	0.71	0.62	0.57
BinanceArabic	0.78	0.72	0.70	0.95	0.93	0.93	0.65	0.54	0.47
ehabtvv	0.75	0.70	0.68	0.96	0.96	0.95	0.66	0.54	0.48
gazaalannet	0.85	0.82	0.79	0.85	0.81	0.78	0.75	0.69	0.65
HARD	0.86	0.80	0.77	0.83	0.77	0.75	0.65	0.56	0.50
jeninBoss	0.83	0.78	0.77	0.80	0.75	0.72	0.71	0.63	0.58
kfoovip	0.77	0.71	0.68	0.93	0.90	0.88	0.68	0.56	0.51
medicinal	0.76	0.68	0.64	0.92	0.89	0.86	0.68	0.57	0.49
MSDA	0.83	0.78	0.76	0.88	0.85	0.83	0.71	0.63	0.57

to those of the clean version, focusing on sentiment accuracy and sensitivity to subtle adversarial perturbations.

4 Experiments and results

In this section, we present and analyze the impact of our dot-level attacks on various sentiment analysis models, including transformer-based models like CamelBERT-Da-Sentiment [11], IMAMU Arabic Sentiment Analysis [2], and its fine-tuned version on the MSDA dataset, as well as traditional deep learning models such as CNN and LSTM. Additionally, we included GraphRAG, a graph-based retrieval-augmented generation model, to assess the resilience of graph-structured LLMs in adversarial sentiment classification.

The effectiveness of our dot-based attack on sentiment analysis tasks using the CAMeL-Lab sentiment model [11] has been evaluated and compared with the character-based Ar-Flip and Ar-Flip2 methods from [3] as shown in Table 2. Our method is capable of flipping a model's sentiment prediction from clearly **positive** (score: 0.927) to strongly **negative** (score: 0.305) using only subtle visual modifications.

Table 4. Attack Success Rate (ASR) of models under different attack levels (L1–L3).

Dataset	**IMAMU**			**Fine-Tuned IMAMU**			**CamelBERT**		
	L1	L2	L3	L1	L2	L3	L1	L2	L3
AlarabyTelevision	19.61	24.78	27.21	18.58	23.07	28.67	28.54	37.85	42.56
BinanceArabic	22.33	27.78	30.08	5.10	6.90	7.39	34.63	46.00	52.85
ehabtvv	24.85	30.18	32.25	3.82	4.32	4.83	34.18	46.13	52.35
gazaalannet	14.67	18.41	20.93	15.06	19.43	21.89	25.33	31.29	35.42
HARD	13.81	20.08	22.77	16.94	22.73	24.99	34.77	44.14	50.34
jeninBoss	17.49	21.63	22.89	20.31	24.74	28.09	28.76	37.40	41.52
kfoovip	23.38	29.22	32.49	7.39	10.25	12.30	32.27	43.65	49.31
medicinal	24.35	31.61	35.59	8.41	11.32	13.69	32.23	42.93	50.72
MSDA	17.44	21.66	24.27	11.86	15.45	17.45	29.34	37.35	42.58

4.1 Performance of pre-trained models under different attack levels

The results presented in Table 3 illustrate the accuracy of the IMAMU, Fine-Tuned IMAMU, and CamelBERT-Da-Sentiment models when tested on different attack levels (L1, L2, and L3) across various datasets. The Fine-Tuned IMAMU model consistently outperforms both the standard IMAMU and CamelBERT-Da-Sentiment models, maintaining higher accuracy across all datasets and attack levels. The accuracy of the Fine-Tuned IMAMU model remains relatively stable, even at higher attack levels, as seen in datasets like BinanceArabic (95% at L1 and 93% at L3) and ehabtvv (96% at L1 and 95% at L3). In contrast, the CamelBERT-Da-Sentiment model demonstrates a significant decline in performance as the attack level increases, with accuracy dropping as low as 47% in BinanceArabic at L3. The CNN-BiGRU-Focus architecture of IMAMU, which combines CNN for local pattern extraction and BiGRU for sequential dependency learning, contributes to its resilience against adversarial perturbations.

While the standard IMAMU model degrades under attack, the Fine-Tuned IMAMU variant remains stable, showing minimal accuracy drops at higher attack levels. CamelBERT rely more heavily on subword tokenization, making them more sensitive to character-level perturbations. Unlike IMAMU, which integrates CNN for feature extraction and BiGRU for sequential processing, transformers encode entire words or subwords, which means minor perturbations can disproportionately affect learned embeddings, leading to larger drops in performance.

Table 4 summarizes ASR values across datasets and attack levels, with higher ASR indicating greater vulnerability. CamelBERT-Da-Sentiment is the most susceptible, exceeding 50% ASR at L3 on datasets like BinanceArabic and ehabtvv. In contrast, the Fine-Tuned IMAMU model shows strong robustness, maintaining ASR below 8% across all levels. The standard IMAMU model demonstrates moderate resilience, highlighting the benefit of fine-tuning in improving adversarial robustness.

4.2 Testing attack on GraphRAG

The responses from the GraphRAG models fine-tuned on adversarial data and clean data reveal significant differences in sentiment analysis and topic identification. For questions related to sentiment analysis (Q1, Q2) in Table 5, the model fine-tuned on adversarial data displayed a more skewed negative sentiment, particularly in regions affected

Table 5. Comparison of responses for adversarial and clean data-fine-tuned GraphRAG models.

Question	Adversarial Data Responses	Clean Data Responses
Q1	Predominantly negative sentiment (70-80%) due to military conflicts, security issues, and political tensions. Neutral sentiment (20-25%) in factual descriptions. Minimal positive sentiment (0-5%) in context.	Negative sentiment (60%) driven by political conflicts and perceived betrayals. Positive sentiment (20%) linked to hope and religious invocations. Neutral sentiment (20%) in factual reporting.
Q2	Ratio skewed towards negative sentiment (3:1), largely due to conflict and military operations. Inferences are made based on context, but no explicit sentiment analysis is provided.	Negative sentiment dominates due to conflicts and security warnings (approximately 2 negative for every 1 positive message).
Q3	Al-Shuja’iyya and Gaza City: Predominantly negative due to military activities, instability, and location importance. Media influence amplifies negativity.	Gaza war movements: Strong negative sentiment due to ongoing struggles. Positive sentiment exists in the form of hope, resilience, and religious support, but is less prevalent.
Q4	The dataset includes reports, entities, and relationships related to conflict zones, military activities, and media channels in Gaza. Negative sentiments from reports on military operations, media influence, and humanitarian concerns.	The dataset allows for sentiment inference. Negative sentiment is prevalent, though some positive messages exist. Positive expressions like alnasr aleazim, which means the great victory, suggest optimism.
Q5	Media channels like the ”Al-Qassam Brigades” Channel could influence sentiment. Influential entities like ”Al-Qassam Brigades” and ”Saraya al-Quds” shape sentiment, especially in conflict zones.	Military movements, religious invocations, and regional conflicts intertwine positive aspirations and negative emotions. Entities like ”Lions of the West Bank” influence sentiment, particularly negative

by military activities. The adversarially trained model produced a higher percentage of negative sentiment (70-80%), driven by factors such as political tensions and security concerns. In contrast, the clean data model showed a more balanced sentiment distribution, with 60% negative sentiment, but also incorporated a higher proportion of positive sentiment (20%) related to hope and religious invocations. These results demonstrate that tuning the model on adversarial data made it more sensitive to negative sentiment in specific contexts, leading to a more polarized response.

For the community sentiment queries (Q3, Q4), the adversarial model identified Gaza and Al-Shuja'iyya as regions with predominantly negative sentiment, influenced by military instability and media narratives. The clean model recognized similar negative sentiment but also highlighted the presence of positive expressions such as Alnasr aleazim (the great victory) within movements in the Middle East. The adversarial model, however, presented a more one-sided negative perspective.

5 Conclusions

In this study, we introduce a novel offensive method to craft adversarial attacks at the graphemic dot-level on Arabic textual content. Our results reveal that even a small number of character substitutions can effectively disrupt the performance of state-of-the-art DNN and pre-trained classifiers for Arabic text and LLM models such as GPT-4o-mini. Moreover, the responses from the GraphRAG models fine-tuned on adversarial and clean data differed significantly, particularly in the intensity of negative sentiment. Our experiments demonstrate that altering just a single character in each sample can lead to a notable decline in classification accuracy while achieving a high ASR. As a future work, we aim to apply this adversarial evaluation technique to a wider range of DNN models across various NLP tasks and language models such as GPT-4-turbo or DeepSeek.

References

1. Abdelaty, M., Lazem, S.: Investigating the robustness of arabic offensive language transformer-based classifiers to adversarial attacks. In: 2024 Intelligent Methods, Systems, and Applications (IMSA) (2024)
2. ALANZI: imamu_arabic_sentimentanalysis. `https://huggingface.co/ALANZI/imamu_arabic_sentimentAnalysis` (2023)
3. Alshemali, B., Kalita, J.: Character-level adversarial examples in arabic. In: 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE (2021)
4. Althafir, Z., Ghnemat, R.: A hybrid approach for auto-correcting grammatical errors generated by non-native arabic speakers. In: 2022 International Conference on Emerging Trends in Computing and Engineering Applications (ETCEA). pp. 1--6 (2022)
5. Baniata, L.H., Kang, S.: Switching self-attention text classification model with innovative reverse positional encoding for right-to-left languages: A focus on arabic dialects. Mathematics 12(6), 865 (2024)
6. Boujou, E., Chataoui, H., Mekki, A.E., Benjelloun, S., Chairi, I., Berrada, I.: An open access nlp dataset for arabic dialects: Data collection, labeling, and model construction. arXiv preprint arXiv:2102.11000 (2021)
7. Chang, G., Gao, H., Yao, Z., Xiong, H.: Textguise: Adaptive adversarial example attacks on text classification model. Neurocomputing 529, 190--203 (2023)
8. Ebrahimi, J., Lowd, D., Dou, D.: On adversarial examples for character-level neural machine translation. arXiv preprint arXiv:1806.09030 (2018)
9. Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S., Metropolitansky, D., Ness, R.O., Larson, J.: From local to global: A graph rag approach to query-focused summarization. arXiv preprint arXiv:2404.16130 (2024)
10. Elnagar, A., Khalifa, Y.S., Einea, A.: Hotel Arabic-Reviews Dataset Construction for Sentiment Analysis Applications, pp. 35--52. Springer International Publishing, Cham (2018)
11. Inoue, G., Alhafni, B., Baimukan, N., Bouamor, H., Habash, N.: The interplay of variant, size, and task type in Arabic pre-trained language models. In: Proceedings of the Sixth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, Kyiv, Ukraine (Online) (Apr 2021)
12. Liu, H., Zhang, Y., Wang, Y., Lin, Z., Chen, Y.: Joint character-level word embedding and adversarial stability training to defend adversarial text. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 8384--8391 (2020)
13. Morris, J.X., Lifland, E., Yoo, J.Y., Grigsby, J., Jin, D., Qi, Y.: Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. arXiv preprint arXiv:2005.05909 (2020)
14. Pruthi, D., Dhingra, B., Lipton, Z.C.: Combating adversarial misspellings with robust word recognition. arXiv preprint arXiv:1905.11268 (2019)
15. Radman, A., Duwairi, R.: Towards a robust deep learning framework for arabic sentiment analysis. Natural Language Processing pp. 1--35 (2022)
16. Salman, A., Alajmi, A., Ahmad, I.: Evaluation of adversarial robustness in arabic language models. Available at SSRN 4954707
17. Zhang, Z., Liu, M., Zhang, C., Zhang, Y., Li, Z., Li, Q., Duan, H., Sun, D.: Argot: Generating adversarial readable chinese texts. In: Bessiere, C. (ed.) Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20. pp. 2533--2539. International Joint Conferences on Artificial Intelligence Organization (7 2020), main track