KFUPM
Journal of Undergraduate Research International

Diamond Open Access 56

Article Computer Science

Identifying Arabic AI-Generated Text on Social Media: Preprocessing Choices and Cross-Platform Generalization

Abdulaziz Alqahtani, Amal Sunba and Tarek Helmy*

Cite https://doi.org/10.64589/juri/231577

Submitted: April 23, 2026 **Revised:** May 12, 2026 **Accepted:** May 14, 2026

ABSTRACT

As large language models have become widely accessible, artificial intelligence (AI)-generated Arabic text designed to mislead readers has become increasingly prevalent on social media, making reliable detection across platforms a challenging task. This study investigates Arabic deepfake text detection across two contrasting environments, Twitter/X and YouTube, using benchmark datasets consisting of paired authentic and AI-generated posts. The effect of text representation was examined by applying three controlled preprocessing views built on a shared cleaning baseline. Within this framework, machine-learning classifiers, sequence-based models, and fine-tuned Arabic transformers were evaluated in both in-domain settings and cross-platform transfer scenarios. Transformer models consistently achieved the strongest in-domain performance, reaching Macro-F1 scores of 0.9912 on Twitter/X and 0.9409 on YouTube. However, the effect of preprocessing was highly variable: retaining the original text generally produced the best results, aggressive manual normalization often degraded performance, and deepfake-aware preprocessing yielded mixed outcomes depending on the model and platform. The cross-platform evaluation revealed a substantial performance gap compared to in-domain results, with the best transfer scores reaching 0.8364 from Twitter/X to YouTube and 0.9433 in the reverse direction. This study provides a controlled cross-platform evaluation of Arabic deepfake text detection and highlights the critical role of preprocessing and model pretraining in shaping robustness. Overall, the findings indicate that while high-accuracy detection within a single platform is achievable, developing models that generalize effectively across platforms remains an open challenge.

Keywords: deepfake, text preprocessing, machine learning, deep learning, transformers, cross-platform transfer

1. INTRODUCTION

The growing use of artificial intelligence (AI) in content generation has made deceptive text an increasingly important challenge in online communication. Deepfake text, unlike manipulated images or videos, can be difficult to detect because it often appears fluent, natural, and contextually appropriate, allowing false or misleading information to blend easily into digital discussions [1,2]. This risk has increased with recent advances in large language models (LLMs) and transformer-based systems, which can generate highly convincing posts, comments, reviews, and articles at scale [3,4]. The problem is especially significant for Arabic, a language that plays an important role in social, political, and public discourse yet remains relatively underexplored in deepfake text detection research [5]. Arabic online content is widely shared on platforms such as Twitter/X and YouTube, where communication patterns differ substantially, with Twitter/X favoring short and fast-moving posts and YouTube supporting longer, more contextual comment threads. In addition, Arabic presents unique linguistic challenges, including rich morphology, dialect diversity, spelling variation, optional diacritics, code-switching, and informal social media usage, all of which complicate reliable text analysis and detection [6–9]. To address related misinformation and AI-text detection tasks, researchers have applied traditional machine learning, deep learning, and transformer-based methods, with pretrained Arabic models, such as AraBERT [10] and MARBERT [11], showing particularly strong performance across several classification tasks. However, despite these advances, important gaps remain, especially in understanding how detection models generalize across platforms, how preprocessing choices influence learned signals, and how well current approaches handle Arabic deepfake text as a distinct and practical detection problem. These challenges underline the need for systematic research that evaluates the robustness of the model across realistic online environments and diverse forms of Arabic discourse.

This study examines Arabic deepfake text detection through a controlled comparison across Twitter/X and YouTube, with the goal of identifying the most effective modeling approaches, understanding the impact of preprocessing and text representation, and evaluating whether models can generalize across platforms without retraining. The main contributions of this study are summarized as follows:

- Comprehensive experimental analysis of previously constructed Arabic deepfake text benchmarks across Twitter/X

Published under an exclusive license by KFUPM

and YouTube, covering paired authentic and AI-generated samples under six deception strategies.
- Systematic comparison of machine learning, sequence-based, and transformer-based approaches under three preprocessing views for Arabic deepfake text detection.
- Comprehensive cross-platform evaluation analyzing model robustness when transferring between Twitter/X and YouTube without retraining.
- Detailed analysis of how preprocessing strategies, model pretraining backgrounds, and deception types influence Arabic deepfake detection performance.

The following research questions guide the study:

- RQ1: Which machine learning, sequence, and transformer configurations achieve the best in-domain performance on Twitter/X and YouTube?
- RQ2: How do preprocessing variants affect in-domain performance across different model families and representation strategies?
- RQ3: How robust are trained models under cross-platform transfer without retraining or target-domain adaptation?
- RQ4: In which deceptive types and platform-specific categories do the strongest models make the most errors?

Based on these research questions, the study assessed the following hypotheses:

- H1: Fine-tuned contextual Arabic transformers will achieve the highest performance on both platforms.
- H2: Preprocessing choices will significantly affect results across model families and representation types.

2. RELATED WORK

Effective detection of Arabic text depends heavily on preprocessing, given the language's inherent complexity, which includes morphological richness, widespread orthographic variation, extensive dialect diversity, and the optional use of diacritics. Prior studies have shown that normalization and cleaning can reduce noise and sparsity, which in turn improve feature learning and model consistency. Dahou et al.[14] applied a multi-step cleaning pipeline that stripped punctuation, stopwords, and diacritics prior to subword tokenization, yielding more consistent input representations and improving the accuracy of their fake-news classifier. Similarly, Alshammari and Elleithy[13] highlighted the importance of managing Arabic orthographic irregularities and showed that removing optional diacritics alone produced a substantial gain in detection accuracy. In addition, Alnabrisi and Saad[15] adopted the official AraBERT preprocessing pipeline, including text cleaning, padding, and subword segmentation, and found that eliminating noisy elements, such as hashtags and user mentions, strengthened model performance. Qandos et al.[16] extended this line of work by demonstrating that effective preprocessing must address dialectal variation, applying expression normalization for both Modern Standard Arabic (MSA) and the Saudi dialect, alongside dialect-specific stopword filtering. Together, these studies confirm that Arabic preprocessing is not a routine preparation step but a critical factor that directly shapes downstream detection quality.

Research on Arabic misinformation and AI-generated text detection has evolved across several main directions. In Arabic text preprocessing, Dahou et al.[14] and Alshammari and Elleithy[13] demonstrated that normalization, diacritic removal, and the cleaning of noisy tokens can substantially improve text representation and detection accuracy, particularly in linguistically diverse Arabic data. In classical machine learning, Alkhair et al.[17] and Alsudias and Rayson[18] demonstrated that models such as support vector machines (SVMs) and logistic regression, combined with term frequency-inverse document frequency (TF-IDF) and frequency-based features, can produce strong results, although they remain limited in capturing deeper contextual information. In deep learning, Wotaifi and Dhannoon[19] and Alshahrani et al.[20] reported that hybrid neural architectures combining convolutional neural network (CNN), long short-term memory (LSTM), and gated recurrent unit (GRU) components improve performance by learning both local and sequential patterns. For transformer-based approaches, Harrag et al.[21] and Al-Yahya et al.[22] found that fine-tuned Arabic transformers, such as AraBERT and QARiB, outperform earlier methods by modeling contextual and semantic information more effectively. Finally, in cross-domain evaluations, Qandos et al.[16] and Houmaine and Rosso[23] showed that although transfer learning and larger multi-domain datasets can improve robustness, cross-domain and cross-platform generalization in Arabic detection remains challenging.

3. METHODOLOGY

3.1. Dataset Construction. The datasets used in this study were previously constructed and evaluated by Alshehri et al.[24] and Alhashem et al.[25], who collected authentic Arabic social-media content from Twitter/X and YouTube and generated corresponding AI-based deceptive variants under controlled deception strategies. Each real text was rewritten using OpenAI o4-mini for Twitter and GPT-4.0 Mini for YouTube to produce a paired AI-generated deepfake counterpart, with six deception patterns applied: exaggeration, contradiction, omission, satire, clickbait, and mixed truth (Table 1). The generated texts were designed as controlled deceptive rewrites rather than direct duplicates, with each sample introducing a targeted semantic modification according to a specific deception strategy while preserving topical consistency. The original dataset studies included data-cleaning procedures and quality evaluation protocols, with random subsets manually inspected to verify linguistic naturalness and non-identical rewriting. Full details of the construction and evaluation are documented in the previous studies[24,25]. These benchmarks were selected because they provided controlled, paired, authentic, and AI-generated Arabic samples across multiple deception categories, making them suitable for systematic preprocessing and cross-platform transfer evaluation. The resulting benchmarks consist of 28,806 texts for Twitter/X and 29,711 texts for YouTube. Twitter/X and YouTube were selected because they represent structurally different forms of Arabic online discourse, enabling the controlled evaluation of model robustness across short informal posts and longer contextual comment threads. Building on these benchmarks, the present study focuses on evaluating preprocessing strategies,

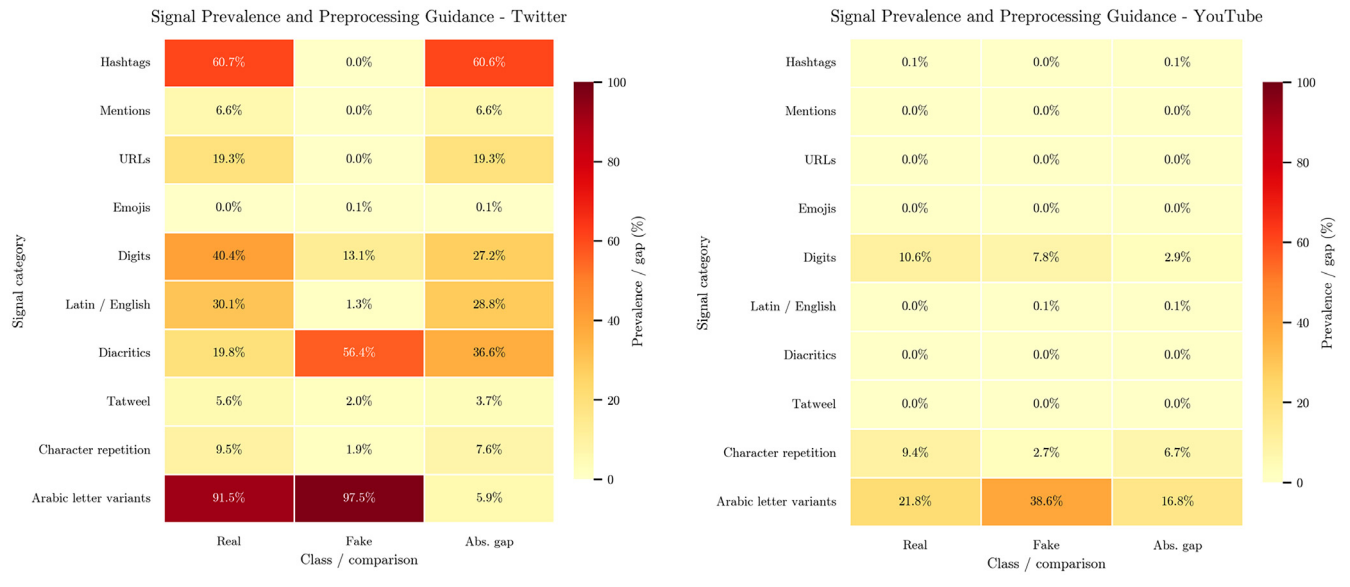
Table 1. *Breakdown of deception sub-types and their example counts on Twitter/X and YouTube*

Deception Type	Description	Count	
		Twitter/X	YouTube
Exaggeration	Amplifies or exaggerates the original claims and emotional tone while keeping the subject matter intact.	2,289	2,543
Omission	Deliberately omits important contextual information, creating a distorted or misleading overall impression.	2,286	2,403
Contradiction	Inserts contradictory assertions that oppose the original meaning yet still appear credible.	2,299	2,406
Satirical Tone	Reframes the content using irony or mockery while retaining a deceptive undertone.	2,376	2,441
Clickbait	Rephrases the message with sensational, emotionally loaded language designed to attract attention.	2,364	2,421
Mixed Truths	Blends accurate facts with fabricated or misleading content to create a superficially plausible narrative.	2,333	2,423
Total	Combined count of generated deceptive examples over all sub-types.	13,947	14,637

comparing multiple model families, and analyzing cross-platform transfer robustness, with the data split using a 70%/20%/10% ratio.

3.2. Preprocessing Strategies. Rather than applying a fixed cleaning pipeline, preprocessing was treated as an experimental variable, and three distinct text views were tested. The first, the original view, preserved the raw text unchanged and served as the baseline. The second, the manual view, applied thorough cleaning by removing URLs, mentions, hashtags, emojis, diacritics, Latin characters, and repeated characters to produce a simplified lexical representation. The third, the deepfake-aware view, adopted a more targeted approach: on Twitter/X, surface markers were replaced with typed placeholders, whereas on YouTube, preprocessing was more conservative to preserve structural cues that may help distinguish between human- and AI-generated text. This view is best understood as an alternative normalization strategy rather than a lighter version of manual cleaning. The rationale behind these choices and the distribution of relevant signals in the training data are illustrated in Figure 1, and the complete per-signal rules are detailed in Table 2. Table 3 presents representative examples illustrating how each view transforms real and AI-generated Arabic tweets, making the effects of each strategy concrete. Text was then tokenized according to model family: whitespace tokenization was used for machine learning and sequence models, whereas WordPiece (maximum sequence length = 128) was used for transformers.

3.3. Modeling Approaches. The three families of models were compared within a unified evaluation framework. For machine learning, SVM, logistic regression, and random forest were trained on TF-IDF features, as well as Word2Vec-CBOW and FastText representations, thereby covering both sparse lexical and dense semantic representations. For sequence-based deep learning, the LSTM and bidirectional LSTM (Bi-LSTM) models processed padded token sequences, with embeddings initialized either randomly or using the same static word vectors, enabling the models to capture sequential dependencies effectively. For



(a) (b)

Figure 1. Distribution of corpus-level signals across the training data: (a) Twitter/X; (b) YouTube

Table 2. *Per-view treatment of each text signal across the three preprocessing pipelines (Twitter/X and YouTube)*

	View 1 (Original)		View 2 (Manual)		View 3 (Deepfake-aware)	
Text Signal	**Twitter/X**	**YouTube**	**Twitter/X**	**YouTube**	**Twitter/X**	**YouTube**
URL links	Preserved	Preserved	Stripped	Stripped	URL placeholder	Stripped
User mentions	Preserved	Preserved	Stripped	Stripped	Mention placeholder	Stripped
Hashtag entries	Preserved	Preserved	Stripped	Stripped	Hashtag placeholder	Stripped
Numeric spans	Preserved	Preserved	As plain text	As plain text	Number placeholder	Number placeholder
Latin/English script	Preserved	Preserved	Stripped	Stripped	Latin placeholder	Stripped
Emoji symbols	Preserved	Preserved	Stripped	Stripped	Stripped	Stripped
Diacritical marks	Preserved	Preserved	Stripped	Stripped	Preserved	Stripped
Alef/Yeh/Hamza	Untouched	Untouched	Enabled	Enabled	Enabled	Disabled
Tatweel	Preserved	Preserved	Stripped	Stripped	Stripped	Stripped
Repeated characters	Preserved	Preserved	Max. 1 repetition	Max. 1 repetition	Max. 2 repetitions	Max. 2 repetitions
Punctuation & separators	Preserved	Preserved	Normalized/ collapsed	Normalized/ collapsed	Normalized/ collapsed	Normalized/ collapsed
Final character filter	Untouched	Untouched	No placeholders	No placeholders	Placeholders kept	Placeholders kept
Whitespace normalization	Preserved	Preserved	Enabled	Enabled	Enabled	Enabled

transformer-based models, three Arabic-specific BERT-based models were fine-tuned on tokenized Arabic text. AraBERTv2 is a BERT-based encoder model pretrained on approximately 77 GB of Arabic text sourced from Wikipedia, news articles, and the OSCAR corpus. It employs Farasa segmentation during preprocessing, making it particularly well-suited for formal MSA. MARBERTv2 was trained on one billion Arabic tweets covering both MSA and dialectal Arabic and was subsequently further pretrained on additional MSA data, making it highly effective for informal social media text, such as Twitter/X. CAMeLBERT is

Table 3. *Sample texts illustrating how the three preprocessing views differ*

	Real Example	**Fake Example**
Text	"شاهد تقرير" @AlArabiya عن ارتفاع أسعار النفط 5% هذا الأسبوع: https://alarabiya.net/economy # اقتصاد	"عاجل! أسعار النفط ترتفع 300% وتهدد الاقتصاد العالمي! تابع @AlArabiya: https://alarabiya.net/fake # اقتصاد
Translation	"Watch @AlArabiya's report on oil prices rising 5% this week: https://alarabiya.net/economy #economy"	"URGENT! Oil prices up 300% threatening the global economy! Follow @AlArabiya: https://alarabiya.net/fake #economy"
View 1: original	Same	Same
View 2: manual	"شاهد تقرير عن ارتفاع اسعار النفط 5% هذا الاسبوع"	"عاجل اسعار النفط ترتفع 300% وتهدد الاقتصاد العالمي"
View 3: deepfake-aware	"شاهد تقرير <MENTION> عن ارتفاع اسعار النفط هذا <NUMBER>% الاسبوع: <URL> <HASHTAG>"	"عاجل اسعار النفط ترتفع وتهدد الاقتصاد <NUMBER>% العالمي تابع <MENTION>: <URL> <HASHTAG>"

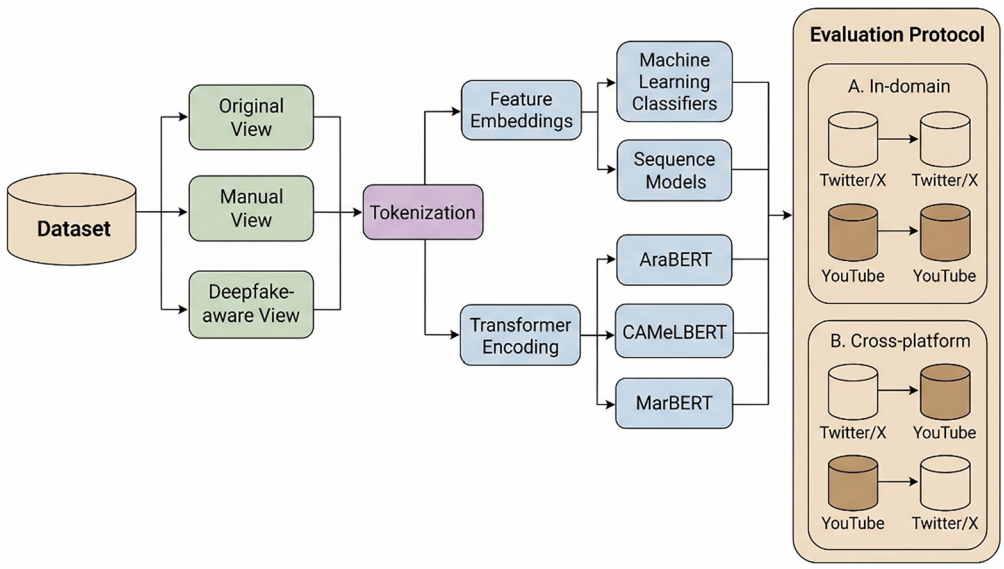


Figure 2. End-to-end view of the study: the three preprocessing pipelines, the model families they feed, and the fixed in-domain and cross-platform tests run on Twitter/X and YouTube

a suite of models pretrained on multiple Arabic varieties, including MSA, dialectal, and classical Arabic, with a mixed variant used in this study to ensure broad coverage across both platforms. These three models were selected to represent distinct pretraining backgrounds: formal Arabic, social media Arabic, and a mix of varieties. Collectively, these model families enable a systematic comparison of representation types and modeling depths across both platforms.

4. EXPERIMENTAL WORK

We kept the evaluation procedure identical for both platforms and for all three preprocessing views, fixing the held-out test partitions in advance and screening out any exact-text leakage between splits before training began. Machine learning models were trained using balanced class weighting, whereas the random forest used 200 trees and a maximum depth of 30. Sequence-based models used a maximum sequence length of 128, an embedding size of 128, a hidden size of 64, and a batch size of 64, and were trained for five epochs using the AdamW optimizer. For the transformers (AraBERTv2, MARBERTv2, and CAMeLBERT), we fine-tuned over five epochs at the same 128-token limit and a batch size of 64, using a learning rate of 2×10^{-5} and a dropout rate of 0.10. The static word vectors were trained separately for 25 epochs at 128 dimensions, and we kept the checkpoint that scored best on validation Macro-F1. As shown in Figure 2, each platform was evaluated under the original, manual, and deepfake-aware text views and assessed using machine learning, sequence, and transformer-based models in both in-domain and cross-platform settings.

5. RESULTS

5.1. In-Domain Performance. For RQ1, both Twitter/X and YouTube yielded the same ordering, with transformer-based models achieving the highest Macro-F1 scores, followed by sequence models and machine-learning classifiers. This ordering is plotted in Figure 6, where each model's performance is reported with 95% confidence intervals. Table 4 summarizes the in-domain Macro-F1 results across the preprocessing views, showing that transformer models consistently achieved the highest family-level averages on both platforms. The strongest overall result was obtained by CAMeLBERT under the original Twitter/X setting, reaching a Macro-F1 score of 0.9912, while MARBERTv2 achieved the best YouTube performance under the deepfake-aware setting, with a Macro-F1 of 0.9369. Across nearly all conditions, transformer models substantially outperformed machine-learning and sequence-based baselines. In relation to RQ2, the results show that preprocessing effects varied across platforms, views, and model families. Manual preprocessing led to the most noticeable decline compared to the original text, while deepfake-aware preprocessing generally reduced this degradation, but did not consistently outperform the original setting. Additionally, Table 5 confirms that the best-performing transformer models significantly outperformed the strongest machine-learning and sequence-model baselines at the family level. Overall, the findings confirm the strong effectiveness of contextual Arabic transformers while also showing that preprocessing effects vary substantially across model families and platforms. Figure 3 shows representative training and validation behavior across all model families. The sequence and transformer models converged rapidly within the first few epochs, while validation performance remained comparatively stable, indicating limited overfitting under the selected training configuration. Machine learning models also maintained stable validation Macro-F1 across configurations, although random forest exhibited larger train–validation gaps than the linear classifiers.

Figures 4 and 5 provide additional comparisons of precision, recall, and F1-score across the best-performing configurations for each model family and preprocessing view on Twitter/X and YouTube, respectively. Transformer-based models maintained the most balanced precision–recall behavior across both platforms, while machine-learning and sequence-based

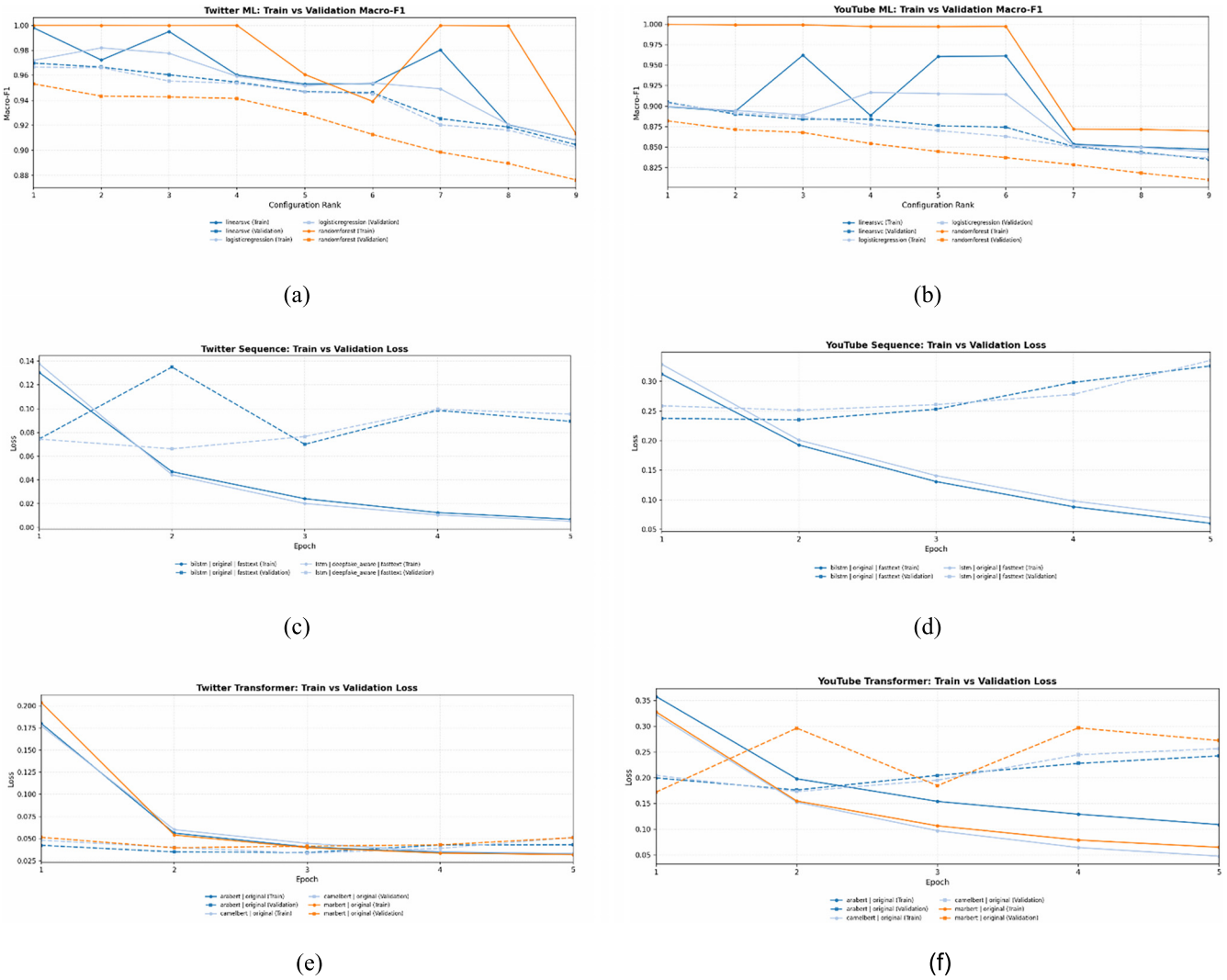


Figure 3. Training and validation performance trends across machine learning, sequence-based, and transformer-based models on the Twitter/X and YouTube benchmark datasets: (a) Twitter/X machine learning Macro-F1, (b) YouTube machine learning Macro-F1, (c) Twitter/X sequence-model loss, (d) YouTube sequence-model loss, (e) Twitter/X transformer loss, and (f) YouTube transformer loss

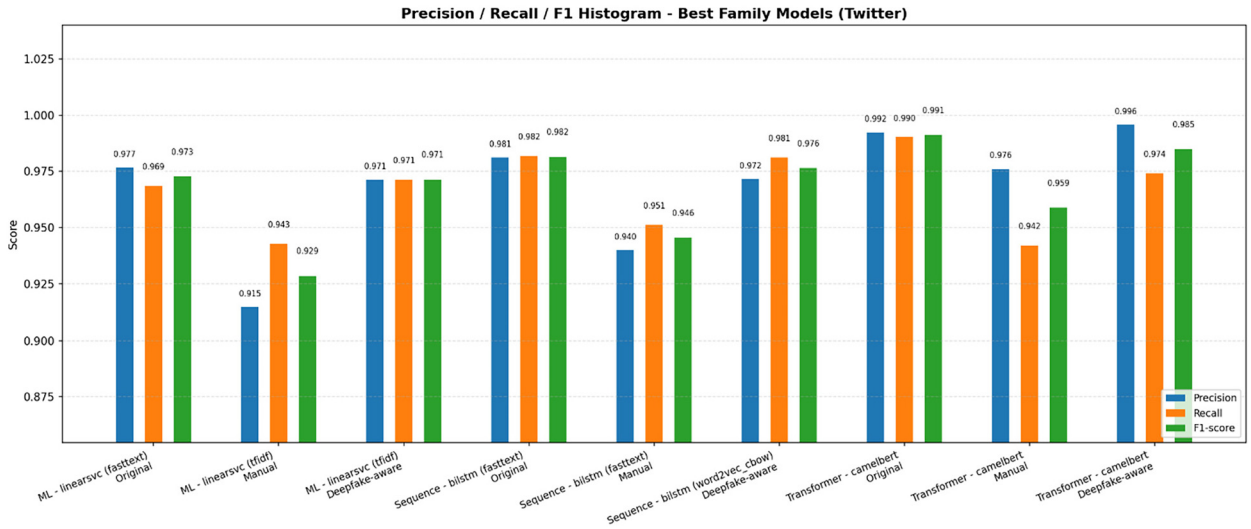


Figure 4. Comparison of precision, recall and F1 scores comparison for the best-performing machine learning, sequence-based, and transformer-based configurations across preprocessing views on the Twitter/X benchmark dataset

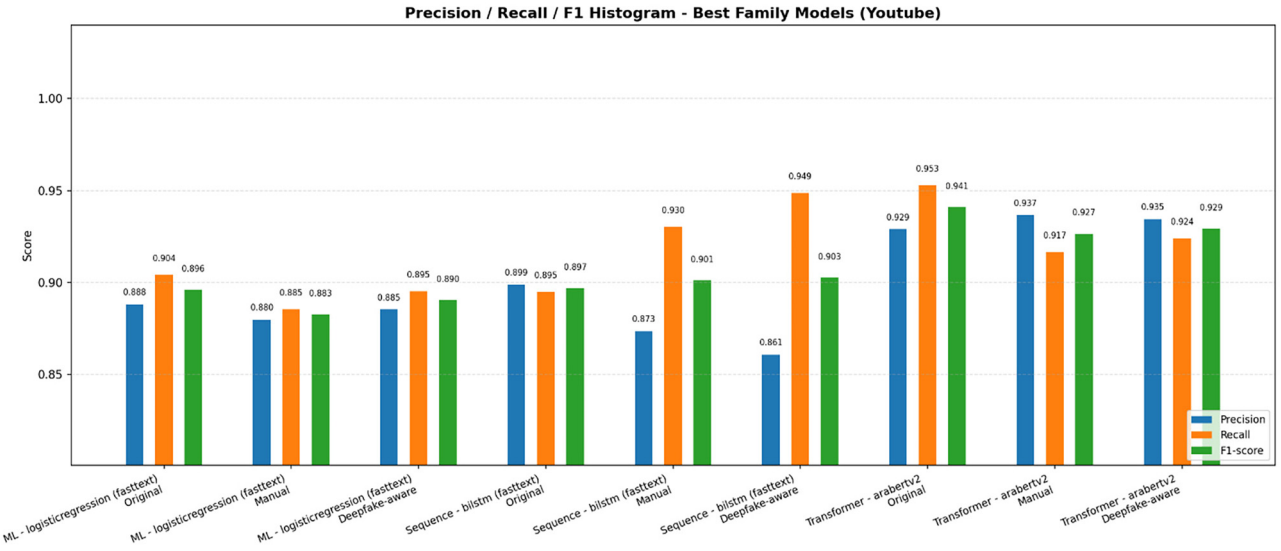


Figure 5. Comparison of precision, recall and F1 scores comparison for the best-performing machine learning, sequence-based, and transformer-based configurations across preprocessing views on the YouTube benchmark dataset

models showed greater variability depending on the representation type and preprocessing strategy. Table 6 summarizes training time (in seconds) across all model families and preprocessing settings. Machine-learning models required the lowest training time overall, with linear classifiers typically completing in well under one second per configuration. Sequence-based models introduced moderate overhead, averaging 12–25 s, depending on architecture and platform. Transformer-based models were substantially more computationally expensive, requiring several hundred seconds per configuration due to contextual encoding and fine-tuning over the full training set. Despite this higher computational cost, transformers consistently achieved the strongest detection performance across both platforms and preprocessing views.

5.2. Error Analysis. According to Table 7, errors are concentrated in specific deception types rather than being evenly distributed, which directly addresses RQ4. On both platforms, omission and contradiction were the most difficult categories for the strongest models, whereas mixed truths remained challenging on YouTube. Taken together, the transformer models incurred the fewest errors, the sequence models fell in between, and the machine-learning models accumulated the most. The results also suggest that preprocessing had a stronger negative effect on Twitter/X than on YouTube.

5.3. Cross-Platform. For RQ3, we probed cross-platform robustness by taking each platform's trained models and running them on the other platform with no further fine-tuning. As listed in Table 8, the cross-platform results were consistently lower than the in-domain performance, although the magnitude of the drop varied depending on the model family and transfer direction. As before, the transformers held up best, the sequence models sat in between, and the machine-learning models suffered the largest transfer penalty. In the Twitter/X-to-YouTube direction, MARBERTv2 with manual preprocessing gave the top score (0.8364), while in the reverse YouTube-to-Twitter/X direction AraBERTv2 under the original view was best (0.9433). The findings also showed that the preprocessing effects were direction-dependent; manual preprocessing was often more effective from Twitter/X to YouTube, whereas the original view was generally more stable from YouTube to Twitter/X. Overall, transfer from YouTube to Twitter/X was more stable, suggesting that YouTube-trained models captured broader patterns, whereas Twitter/X-trained models relied more on platform-specific signals. Transformer models demonstrated clear superiority over both sequence and machine-learning baselines in the transfer setting, with AraBERTv2 achieving above 0.90 Macro-F1 across all three preprocessing views in the YouTube-to-Twitter/X direction. Machine learning models showed the steepest degradation under cross-platform transfer, particularly under deepfake-aware

Table 4. *In-domain Macro-F1 performance across preprocessing views, including family-level averages and best-performing configurations*

Platform	Preprocessing View	Machine Learning mean	Sequence Models mean	Transformers mean	Worst Model	Best Model
Twitter/X	Original	0.9581	0.9757	0.9906	RF-TFIDF (0.9135)	CAMeLBERT (0.9912)
	Manual	0.9093	0.9344	0.9596	RF-TFIDF (0.8770)	MARBERTv2 (0.9625)
	Deepfake-aware	0.9548	0.9736	0.9857	RF-TFIDF (0.9324)	MARBERTv2 (0.9869)
YouTube	Original	0.8678	0.8898	0.9390	RF-TFIDF (0.8297)	MARBERTv2 (0.9409)
	Manual	0.8587	0.8826	0.9303	RF-TFIDF (0.8225)	MARBERTv2 (0.9354)
	Deepfake-aware	0.8645	0.8888	0.9318	RF-TFIDF (0.8250)	MARBERTv2 (0.9369)

Table 5. *Statistical significance tests at the model-family level comparing top-performing transformers against the best machine learning and sequence-based baselines on each platform*

Platform	Comparison Family	Transformer Model	Comparison Model	p-value	p-adjusted
Twitter/X	Machine Learning	CAMeLBERT	SVM	1.55×10^{-10}	4.66×10^{-10}
	Sequence	CAMeLBERT	Bi-LSTM	3.0×10^{-4}	5.0×10^{-4}
YouTube	Machine Learning	MARBERTv2	Logistic Regression	1.26×10^{-27}	3.78×10^{-27}
	Sequence	MARBERTv2	Bi-LSTM	2.47×10^{-25}	3.7×10^{-25}

preprocessing in the Twitter/X-to-YouTube direction, where several configurations dropped below 0.55. Preprocessing sensitivity also differed markedly between transfer directions; manual preprocessing consistently improved Twitter/X-to-YouTube transfers for most model families, whereas the original view was generally more stable in the reverse direction.

6. DISCUSSION

The cross-platform experiments revealed that model robustness depends not only on architecture but also on platform structure, preprocessing strategy, and linguistic characteristics captured during pretraining. Twitter/X posts are short, hashtag-dense, and informal, meaning that models trained on Twitter/X text learn to rely on surface-level lexical signals that are largely absent from YouTube comments. When such models are applied to YouTube, these signals disappear, leading to a performance drop unless preprocessing bridges this gap. This explains why manual preprocessing, by standardizing informal markers, improves Twitter/X-to-YouTube transfer. Conversely, YouTube comments are longer and more varied in register; therefore, models trained on YouTube data encounter a richer distributional signal that partially overlaps with Twitter/X text, even without normalization. This results in a more stable YouTube-to-Twitter/X transfer, as shown in Table 8.

The stronger cross-platform performance of AraBERTv2 from YouTube to Twitter/X likely reflects its pretraining on formal MSA news and encyclopedic text: when applied to the relatively structured YouTube corpus during training, AraBERTv2 learns generalizable linguistic patterns that transfer to Twitter/X more effectively than models relying on dialect-heavy or tweet-specific representations. The advantage of MARBERTv2 in the Twitter/X-to-YouTube manual condition is consistent with its broad dialectal pretraining; after manual normalization

Table 6. *Training time comparison across machine learning, sequence-based, and transformer-based models under different preprocessing views on the Twitter/X and YouTube benchmark datasets*

			Twitter/X			YouTube		
Family	Model	Embedding	Original (s)	Manual (s)	Deepfake-aware (s)	Original (s)	Manual (s)	Deepfake-aware (s)
Machine Learning	SVM	TF-IDF	0.40	0.37	0.33	0.25	0.23	0.24
	Logistic Regression	TF-IDF	0.13	0.13	0.12	0.07	0.07	0.13
	Random Forest	TF-IDF	1.23	1.33	1.24	0.82	0.82	0.89
	SVM	Word2Vec-CBOW	1.34	1.97	1.32	4.21	4.72	4.56
	Logistic Regression	Word2Vec-CBOW	1.48	1.91	1.61	1.63	1.65	1.74
	Random Forest	Word2Vec-CBOW	8.65	8.63	8.03	6.77	7.06	6.74
	SVM	FastText	0.68	0.96	0.75	0.77	0.71	0.79
	Logistic Regression	FastText	0.77	0.82	0.71	0.55	0.49	0.49
	Random Forest	FastText	8.91	9.13	8.13	7.11	6.57	6.96
Sequence	LSTM	Random	17.67	16.51	16.66	12.42	11.86	11.90
	LSTM	Word2Vec-CBOW	16.59	16.71	16.59	12.07	11.97	11.97
	LSTM	FastText	16.46	17.05	17.02	11.89	11.93	11.94
	Bi-LSTM	Random	25.46	25.40	25.38	17.88	18.06	18.09
	Bi-LSTM	Word2Vec-CBOW	24.91	25.37	25.09	18.10	18.10	18.10
	Bi-LSTM	FastText	24.93	25.31	25.08	17.85	17.92	18.12
Transformer	AraBERTv2	Contextual	425.33	424.35	424.09	310.77	310.44	310.61
	MARBERTv2	Contextual	425.13	424.71	424.39	317.42	317.44	317.36
	CAMeLBERT	Contextual	393.07	392.66	392.75	290.74	290.57	290.87

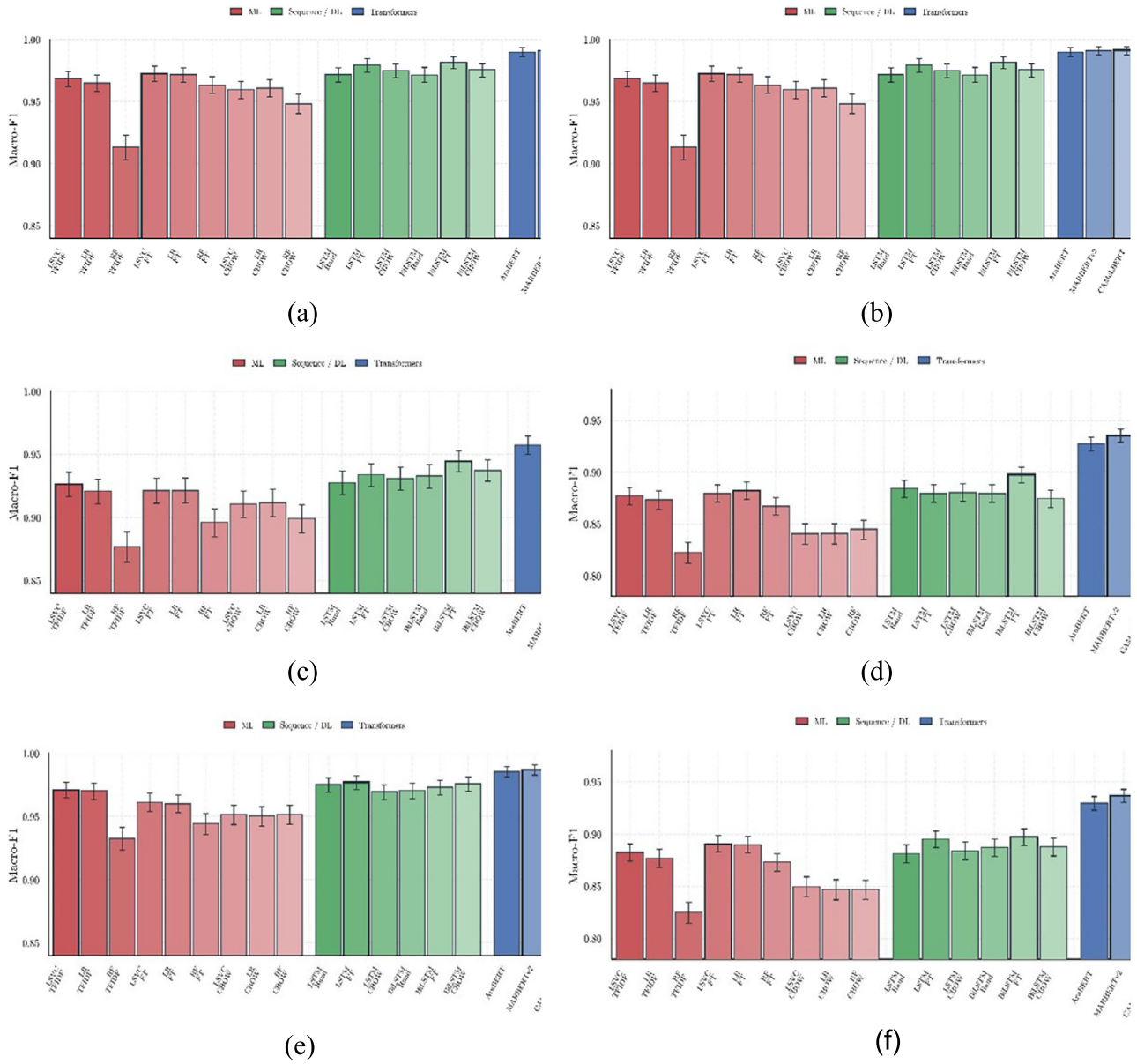


Figure 6. In-domain Macro-F1 scores with 95% confidence intervals across preprocessing views: (a) Twitter/X–original, (b) YouTube–original, (c) Twitter/X–manual, (d) YouTube–manual, (e) Twitter/X–deepfake-aware, (f) YouTube–deepfake-aware

reduces Twitter/X-specific noise, MARBERTv2 benefits from its exposure to diverse Arabic varieties, providing a wider vocabulary overlap with YouTube content. The stronger performance of CAMeLBERT under the original and deepfake-aware settings may be attributed to its mixed pretraining across multiple Arabic varieties, which enables it to retain useful contextual cues that remain present in less normalized text. As the original and deepfake-aware views preserve more surface-level structures and platform-specific markers, CAMeLBERT benefits more from these heterogeneous signals than models trained primarily on formal MSA or tweet-specific data.

The underperformance of most models under deepfake-aware preprocessing in the Twitter/X-to-YouTube direction suggests that deepfake signals are platform-specific. Markers predictive of AI generation on Twitter/X may not generalize to the different stylistic norms of YouTube, and the preprocessing step may inadvertently remove cues that would have aided transfer. Taken together, these findings suggest that contextual Arabic transformers are the most reliable choice for in-domain deepfake text detection in this setting. However, cross-platform deployment

Table 7. *Mean error rates of the exact-match models across deception categories*

Deception Type	Twitter/X Mean Error (%)	YouTube Mean Error (%)
Original (real texts)	4.3	10.0
Clickbait phrasing	0.5	3.8
Contradiction	8.2	24.7
Exaggeration	1.6	7.1
Mixed truths	1.9	13.2
Omission	17.7	24.4

Table 8. *Macro-F1 scores for cross-platform transfer in both directions between Twitter/X and YouTube. The top score in each column is highlighted in bold*

			Twitter/X to YouTube			YouTube to Twitter/X		
Family	**Model**	**Embedding**	**Original**	**Manual**	**Deepfake-aware**	**Original**	**Manual**	**Deepfake-aware**
Machine Learning	SVM	TF-IDF	0.6405	0.7746	0.5168	0.7453	0.7584	0.7552
	Logistic Regression	TF-IDF	0.6823	0.7861	0.5491	0.7290	0.7472	0.7313
	Random Forest	TF-IDF	0.6211	0.6633	0.7550	0.6672	0.7212	0.6845
	SVM	Word2Vec-CBOW	0.6602	0.7594	0.7321	0.6287	0.6705	0.6284
	Logistic Regression	Word2Vec-CBOW	0.6624	0.7583	0.7344	0.6250	0.6584	0.6243
	Random Forest	Word2Vec-CBOW	0.6590	0.7517	0.7204	0.6456	0.6554	0.6485
	SVM	FastText	0.6904	0.7614	**0.7613**	0.5839	0.6724	0.5690
	Logistic Regression	FastText	0.6804	0.7554	0.7607	0.5823	0.6710	0.5688
	Random Forest	FastText	0.5990	0.7032	0.7155	0.7415	0.7700	0.6024
Sequence	LSTM	Random	0.6369	0.7486	0.6350	0.5450	0.6335	0.4612
	LSTM	Word2Vec-CBOW	0.6490	0.7550	0.6191	0.6566	0.6316	0.5171
	LSTM	FastText	0.6388	0.7317	0.5720	0.6353	0.6996	0.7030
	Bi-LSTM	Random	0.6211	0.7780	0.6506	0.7358	0.7129	0.5747
	Bi-LSTM	Word2Vec-CBOW	0.6569	0.8048	0.6319	0.4983	0.5994	0.5842
	Bi-LSTM	FastText	0.6434	0.7641	0.6454	0.6391	0.6840	0.5299
Transformer	AraBERTv2	Contextual	0.7356	0.8319	0.6582	**0.9433**	**0.9070**	**0.9099**
	MARBERTv2	Contextual	0.7195	**0.8364**	0.6845	0.9079	0.8823	0.9095
	CAMeLBERT	Contextual	**0.7461**	0.7760	0.7206	0.9090	0.8753	0.8372

likely requires platform-aware fine-tuning or domain adaptation to maintain robust performance.

7. CONCLUSIONS

Fine-tuned Arabic transformer models consistently outperformed sequence-based and machine-learning approaches on both platforms, with the best in-domain Macro-F1 reaching 0.9912 on Twitter/X and 0.9409 on YouTube. The impact of preprocessing varied considerably across model families and platforms: aggressive manual cleaning often eliminated signals that aided detection, while the deepfake-aware strategy partially mitigated this effect yet still failed to surpass unprocessed text in every configuration. Applying models across platforms proved far harder than evaluating them within the same platform, with the highest achieved transfer scores being 0.8364 in the Twitter/X-to-YouTube direction and 0.9433 in the reverse. Error analysis indicated that omissions and contradictions were the most challenging deceptive categories on both platforms.

This study contributes a systematic cross-platform evaluation, demonstrating that strong in-domain performance does not guarantee reliable transfer, and that preprocessing choices and transformer pretraining backgrounds jointly shape detection robustness. The practical implication is that contextual Arabic transformers are the strongest choice for in-domain settings; however, cross-platform deployment requires platform-aware adaptation strategies.

Despite these contributions, this study was limited to two benchmark datasets and a controlled single-model generation pipeline, where each platform relied on outputs generated by one LLM. As a result, some detected patterns may reflect characteristics specific to these models rather than universal properties of Arabic deepfake text. Although the findings are expected to be partially generalizable to other instruction-tuned LLMs that produce fluent Arabic text, performance may vary across models with different linguistic styles, prompting strategies, or generation behaviors. Therefore, future work should evaluate additional Arabic platforms and more diverse generative models to better assess cross-model generalization.

ACKNOWLEDGEMENTS

This study was conducted as a collaboration between the General Authority for Defense Development (GADD) and King Fahd University of Petroleum & Minerals (KFUPM), through the Interdisciplinary Research Center for Intelligent Secure Systems under Agreement Number CSSP2643, in Saudi Arabia. The authors acknowledge the General Authority for Defense Development (GADD) in Saudi Arabia for funding this research through project number GADD_2024_01_438. The authors also acknowledge the contributors responsible for the data collection and deepfake text generation conducted in a prior project, whose efforts were essential to this study.

AFFILIATIONS AND AUTHOR DETAILS

Undergraduate Authors

Abdulaziz Alqahtani – *Department of Information and Computer Science, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia*; 0009-0001-6284-1214
Email: s202283240@kfupm.edu.sa

Amal Sunba – *PhD Candidate, Department of Information and Computer Science, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia*; 0009-0002-4712-2104
Email: g202203780@kfupm.edu.sa

Corresponding Author

Tarek Helmy – *Research Mentor, Professor, Department of Information and Computer Science, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia*; 0000- 0001-9683-682X
Email: helmy@kfupm.edu.sa

REFERENCES

(1) Fisher, S. A., Howard, J. W. & Kira, B. Moderating synthetic content: The challenge of generative AI. Philos. Technol. **37**, 133 (2024).

(2) Shoaib, M. R., Wang, Z., Ahvanooey, M. T. & Zhao, J. Deepfakes, misinformation, and disinformation in the era of frontier AI, generative AI, and large AI models. *Proc. Int. Conf. Comput. Appl.*, Cairo, Egypt, 1–7 (2023).

(3) Georgiou, G. P. Differentiating between human-written and AI-generated texts using automatically extracted linguistic features. *Information* **16**, 979 (2025).

(4) Ippolito, D., Duckworth, D., Callison-Burch, C. & Eck, D. Automatic detection of generated text is easiest when humans are fooled. *Proc. Annu. Meet. Assoc. Comput. Linguist.* **58**, 1808–1822 (2020).

(5) Himdi, H., Weir, G., Assiri, F. & Al-Barhamtoshy, H. Arabic fake news detection based on textual analysis. *Arab. J. Sci. Eng.* **47**, 10453–10469 (2022).

(6) Muaad, A. Y., Raza, S., Naseem, U. & Davanagere, H. J. J. Arabic text detection: A survey of recent progress, challenges and opportunities. *Appl. Intell.* **53**, 29845–29862 (2023).

(7) Elnagar, A., Yagi, S. M., Nassif, A. B., Shahin, I. & Salloum, S. A. Systematic literature review of dialectal Arabic: Identification and detection. *IEEE Access* **9**, 31010–31042 (2021).

(8) Alsuwaylimi, A. A. Arabic dialect identification in social media: A hybrid model with transformer models and BiLSTM. *Heliyon* **10**, e36280 (2024).

(9) Inoue, G., Alhafni, B., Baimukan, N., Bouamor, H. & Habash, N. The interplay of variant, size, and task type in Arabic pre-trained language models. *Proc. Arab. Nat. Lang. Process. Workshop*, Kyiv, Ukraine, 92–104 (2021).

(10) Antoun, W., Baly, F. & Hajj, H. AraBERT: Transformer-based model for Arabic language understanding. *arXiv* 2003.00104 (2020).

(11) Abdul-Mageed, M., Elmadany, A. & Nagoudi, E. M. B. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. *Proc. Annu. Meet. Assoc. Comput. Linguist. Int. Joint Conf. Nat. Lang. Process.* **59**, 7088–7105 (2021).

(12) Albtoush, E. S., Gan, K. H. & ALrababah, S. A. A. Evaluation of machine learning and deep learning models for fake news detection in Arabic headlines. *IEEE Access* **13**, 162009–162026 (2025).

(13) Alshammari, H. & Elleithy, K. Toward robust Arabic AI-generated text detection: Tackling diacritics challenges. Information **15**, 419 (2024).

(14) Dahou, A., Ewees, A. A., Hashim, F. A., Al-qaness, M. A. A., Orabi, D. A., Soliman, E. M., Tag-eldin, E. M., Aseeri, A. O. & Abd Elaziz, M. Optimizing fake news detection for Arabic context: A multi-task learning approach with transformers and an enhanced nutcracker optimization algorithm. *Knowl.-Based Syst.* **280**, 111023 (2023).

(15) Alnabrisi, I. K. & Saad, M. K. Detect Arabic fake news through deep learning models and transformers. *Expert Syst. Appl.* **251**, 123997 (2024).

(16) Qandos, N., Hamad, G., Alharbi, M., Alturki, S., Alharbi, W. & Albelaihi, A. A. Multiscale cascaded domain-based approach for Arabic fake reviews detection in e-commerce platforms. *J. King Saud Univ. Comput. Inf. Sci.* **36**, 101926 (2024).

(17) Alkhair, M., Meftouh, K., Smaïli, K. & Othman, N. An Arabic corpus of fake news: Collection, analysis, and classification. In *Arabic Language Processing: From Theory to Practice* (ed. Smaïli, K.) 292–302 (Springer, Cham, 2019).

(18) Alsudias, L. & Rayson, P. COVID-19 and Arabic Twitter: How can Arab world governments and public health organizations learn from social media? *Proc. Workshop NLP COVID-19 ACL*, Online (2020).

(19) Wotaifi, T. A. & Dhannoon, B. N. An effective hybrid deep neural network for Arabic fake news detection. *Baghdad Sci. J.* **20**, Article 20 (2023).

(20) Alshahrani, H. J., Hassan, A. Q. A., Tarmissi, K., Mehanna, A. S., Motwakel, A., Yaseen, I., Abdelmageed, A. A. & Eldesouki, M. I. Hunter prey optimization with hybrid deep learning for fake news detection on Arabic corpus. *Comput. Mater. Contin.* **75**, 4255–4272 (2023).

(21) Harrag, F., Dabbah, M., Darwish, K. & Abdelali, A. BERT transformer model for detecting Arabic GPT-2 auto-generated tweets. *Proc. Arab. Nat. Lang. Process. Workshop* 207–214 (2020).

(22) Al-Yahya, M., Al-Khalifa, H., Al-Baity, H., AlSaeed, D. & Essam, A. Arabic fake news detection: Comparative study of neural networks and transformer-based approaches. *Complexity* **2021**, 5516945 (2021).

(23) Houmaine, A. & Rosso, P. MAGENTA+: Detection of Arabic machine-generated text across domains. in Int. Conf. Artif. Intell. Secur. Netw. (eds. Lazaar, M., Fakhri, Y. & El Makrani, A.) 633–648 (Springer Nature Switzerland, Cham, 2026).

(24) Alshehri, B., Sunba, A. & Helmy, T. Building and evaluating an Arabic social-media dataset for deepfake text detection. J. Undergrad. Res. Int. (2026).

(25) Alhashem, M., Sunba, A. & Helmy, T. Arabic reviews dataset for deepfake text detection: Collection and generation. J. Undergrad. Res. Int. (2026).