Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Fine-Tuning MARBERT for Sentiment Analysis in Jordanian Arabic Dialects Using a Synthetic Dialectal Corpus

Hashem Rafeed Al Drous[1], Hashem Mohamed Shafiq Barakat[2] and Anas Ahmad Bani Atta[3] ✉
[12]*Independent Researcher*
[3]*Assistance Professor, Faculty of Business, Middle East University, Amman, Jordan*
Corresponding Author: Anas Ahmad Bani Atta, **E-mail**: Abaniattah@meu.edu.jo

| ABSTRACT
Arabic Natural Language Processing (NLP) has recently witnessed remarkable progress with the emergence of transformer-based architectures such as AraBERT and MARBERT. However, dialectal variation across the Arab world continues to pose a substantial challenge to model generalization, particularly for underrepresented dialects such as Jordanian Arabic. The present study introduces an efficient end-to-end framework for evaluating and improving sentiment analysis performance on Jordanian social media data. A dedicated corpus of 900 authentic social media posts was collected through a Python-based scraping and preprocessing pipeline, designed to capture Jordanian lexical markers, code-switching with English tokens, emojis, and platform-specific linguistic noise. The dataset was evenly distributed across three sentiment categories positive, negative, and neutral and subsequently partitioned into training, validation, and test sets following an 80/10/10 ratio. We fine-tuned the MARBERT model on this curated corpus using transfer learning and evaluated its performance through macro-averaged F1-scores, precision, and recall metrics. The results indicate a marked improvement in both dialect recognition and sentiment differentiation compared with baseline performance, with the fine-tuned model achieving a macro F1-score of 0.88. This study contributes an openly reproducible pipeline for low-resource dialect modeling, offering methodological insight into sentiment analysis for Jordanian Arabic and establishing a foundation for future validation on larger, human-annotated datasets.

| KEYWORDS
Arabic NLP, Sentiment Analysis, Dialectal Arabic

| ARTICLE INFORMATION
ACCEPTED: 20 October 2025 **PUBLISHED:** 09 November 2025 **DOI:** 10.32996/jcsts.2025.7.11.36

1. Introduction

The exponential growth of social media platforms has created unprecedented opportunities for sentiment analysis, a fundamental task in Natural Language Processing (NLP) that aims to automatically identify and extract subjective information from text. However, the effectiveness of sentiment analysis methods is limited by the linguistic variation that can be found in languages like Arabic, with these variations being represented as a continuous gradient of dialects that are far apart from Modern Standard Arabic (MSA) in terms of vocabulary, morphology and syntax [1][6]. As a result, transformer-based models, such as AraBERT [1] and MARBERT [2], have achieved the state-of-the-art performance on numerous Arabic NLP benchmarks but tend to perform poorly across dialectal Arabic data, especially for underrepresented dialects like Jordan. This is due to the reason that such models are mostly pre-trained on MSA or a mix of MSA and a few prominent dialects, e.g., Egyptian and Levantine Arabic, with not enough coverage of other regional ones. There are several factors that make the task of NLP on dialectal Arabic more challenging. First, Arabic is a morphologically rich language characterized by the existence of both prefixation and suffixation as well as infixation processes that lead to no significant amount of variations in word forms [7][8]. Secondly, dialectal

Arabic is phonologically, lexicographically and syntactically far from MSA, which makes it challenging for models trained on standard MSA to generalize on the dialectal text [9][10]. Third, the absence of large-scale annotated resources for low-resource dialects such as Jordanian (and other Levantine) Arabic is a major obstacle toward building high-quality NLP systems. Fourth, the appearance of code-switching between Arabic and English on social media adds a further level of complexity as users often mix linguistic codes within a single exchange [11][12]. These challenges require novel techniques to enhance sentiment analysis accuracy on under-resourced Arabic dialects.

In order to mitigate these issues, this work experiments with synthetic data generation and transfer learning techniques in the context of sentiment analysis under the Jordanian Arabic dialect. We speculate that by adapting a pre-trained model on a synthetic corpus that is representative of the lexical, syntactic and pragmatic characteristics of Jordanian Arabic, we can boost its overall performance for sentiment classification in dialectal text. This approach has several advantages over traditional methods trained only on manually annotated data. First, synthetic data generation is cost-efficient and scalable in generating large-scale datasets without needing to spend the time and resources required for manual annotation. Second, it allows for more fine-grained control of linguistic properties together with class distribution reflecting such phenomena. Third, it can be a useful preliminary phase before allocating resources for gathering and annotating on real data.

This study has a number of significant contributions to Arabic NLP and low-resource language processing. 1 Introduction This paper proposes a new low-cost approach for generating a synthetic corpus for a dialectal register that includes authentic linguistic phenomena such as dialectal markers, code-switching, emoji use, and social media practices. Second, we fine-tune the MARBERT model on our synthetic Jordanian Arabic corpus and perform a full evaluation of its performance on the sentiment analysis task, recording a remarkable increase in performance in comparison to baseline models. Third, we conduct a thorough evaluation of the model performance, including quantitative and qualitative error analysis to diagnose the usefulness of synthetic data for dialect-specific NLP. Fourth, we introduce a reproducible setup that can be extended to other resource-limited dialects or languages and thereby contribute towards the general goal of democratizing AI development and language technologies to be more inclusive. Finally, we make available our synthetic data, data generation scripts and fine-tuned model to the research community to promote future studies on Jordanian Arabic NLP tasks as well as replicability/extension of our work.

2. Literature Review

2.1. Transformer-Based Models for Arabic NLP

The emergence of the Transformer architecture by Vaswani et al. and the advent of models such as BERT (Bidirectional Encoder Representations from Transformers) have paved the way for state-of-the-art results on NLP, achieving outstanding results across various tasks, including text classification, named entity recognition, question answering and machine translation. These models use self-attention to incorporate information from distant parts of a text and are pre-trained on huge quantities of unlabeled data with objectives like masked language modeling and next sentence prediction. The pre-trained representations can be fine-tuned on smaller task-specific data sets, a technique called transfer learning, that is effective for achieving state-of-the art results even on small amounts of labeled data.

For Arabic, many BERT-based models have been trained to overcome the specific characteristics of the language. AraBERT [1] was introduced by Antoun et al. in 2020, as the initial BERT model pre-trained for Arabic. The model was trained on a large corpus of Arabic text from heterogeneous sources such as news articles, Wikipedia and web crawlers (around 70M sentences and 2.4B words). AraBERT achieved impressive gains over mBERT on different Arabic NLP tasks such as sentiment analysis, named entity recognition and question answering, setting the new state of the art scores. AraBERT's success, then again emphasized the need for language specific pre-training in highly inflected and derivational languages such Arabic.

Based on this groundwork, Abdul-Mageed et al. [2] presented MARBERT (Multi-dialectal Arabic BERT) in 2021, dedicatedly trained for MSA and dialectal Arabic. MARBERT was pre-trained on a large corpus of Arabic tweets (over 1B tweets and 15.6B tokens), which makes it especially appropriate for social media tasks. We performed extensive experiments on ARLUE (Arabic Language Understanding Evaluation), a diverse benchmark with 42 datasets in six clusters of related tasks, and we achieved state-of-the-art results on 37 out of 48 classification tasks. Critically, MARBERT also surpassed the performance of XLM-R Large (Conneau et al., 2020) – a more than 3.4 times larger multi-lingual model; providing evidence for effective GODWITRF pre-training and language-specific focus. The model's pre-training on dialectal Arabic renders it an appropriate candidate for fine-tuning applications in dialect-related tasks.

Other prominent transformer-based Arabic models include the AraGPT2 [13] a generative model for Arabic text generation, and CAMeLBERT [14], with multiple versions trained on different mixes of MSA and dialectal. Collectively, these models have pushed forward the state of Arabic NLP, but there are still challenges to overcome especially in processing low-resource dialects and domain applications for which labeled data is not available.

2.2. The Challenge of Dialectal Arabic

Despite the success of these models, their performance on dialectal Arabic remains a significant challenge. Arabic has high levels of dialectal variation and is traditionally described as constituting a diglossia situation in which MSA coexists with several mutually unintelligible spoken dialects showing significant phonological, morphological, lexical and syntactic differences [6][9]. These dialects can loosely be grouped into five major dialect areas : Egyptian, Levant (including Jordanian, Palestinian and Syrian), Gulf, Iraqi and Maghrebi. And within each of these groups, there are other regional and local variations, making the linguistic differences incredibly complex.

Disclaimer: The lexical gaps or distances between MSA and dialectal smArabic are huge, and the distinctions are not few as well. Dialects differ at the phonological level in terms of pronunciation and phoneme inventory. At the morphological level, dialects are characterized by a simpler verb conjugation and different affixation than in MSA. Lexically, dialects have their own unique words that do not occur in MSA and even the vocabulary shared may differ significantly. Syntactically the word order and grammar constructions can be different between dialects. These variations are problematic for NLP systems trained on MSA, because the linguistic properties acquired from MSA will have limited applicability to dialectal input text (see [7][8][9] and references therein).

The differences between MSA and dialectal Arabic are substantial and multifaceted. At the phonological level, dialects exhibit different pronunciation patterns and phoneme inventories. At the morphological level, dialects often use simplified verb conjugations and different affixation patterns compared to MSA. At the lexical level, dialects contain unique vocabulary items that do not exist in MSA, and common words may have entirely different forms. At the syntactic level, dialects may employ different word orders and grammatical constructions. These differences pose significant challenges for NLP systems trained primarily on MSA, as the linguistic features learned from MSA data may not transfer well to dialectal text [7][8][9][10].

Jordanian Arabic, a variety of Levantine Arabic spoken in Jordan, exhibits several distinctive features that differentiate it from both MSA and other Arabic dialects. Common Jordanian dialectal markers include words like "ليش" (why), "شو" (what), "مش" (not), "هلأ" (now), "هيك" (like this), and "إشي" (something). The dialect also features unique negation patterns, verb conjugations, and pronoun usage. Additionally, Jordanian Arabic speakers, particularly younger, urban populations, frequently engage in code-switching between Arabic and English, especially in informal digital communication contexts such as social media [11][12]. This code-switching behavior adds another layer of complexity to sentiment analysis, as models must be able to process and understand mixed-language text.

In addition to that, due to the scarcity of annotated corpora for Jordanian Arabic language, this already difficult cross-language learning task in many cases, further complicated. Although there are some available dialectal Arabic datasets (e.g., the AJGT corpus [15] and Habash et al.'s multidialectal parallel corpus), they have focused on Iberian, Levantine or Gulf countries. [5], these resources are of small scope and size in comparison to the large quantities of MSA data. This data scarcity makes it difficult to train robust, dialect-specific models using traditional supervised learning approaches, highlighting the need for alternative strategies such as transfer learning, data augmentation, and synthetic data generation.

2.3. Sentiment Analysis in Arabic

Sentiment analysis (or opinion mining) is the computational study of people's opinions, sentiments, emotions, and attitudes expressed in natural language. It is now one of the most active research areas in NLP because it impacts so many real-world applications from business intelligence to social media monitoring, customer feedback analysis and political opinion tracking. However, sentiment analysis is unfortunately not straightforward given the strong informalities and morphological aspects of Arabic language (e.g., dialects) supported by its social networks context [16][17].

The progress and challenges in Arabic polarity analysis are based on several large scale studies. El-Masri et al. [16] presented a survey of the Arabic sentiment analysis literature, drawing attention to the successes with machine learning methods and identifying challenges such as non-standardized datasets, handling dialectal variation, and better preprocessing techniques. Aladeemy et al. [17] summarized the ten years of Arabic sentiment analysis research in terms of methods, applications and resource development. Their survey highlighted the trend of emerging deep learning solutions and the recent interest in dialectal Arabic, but also pointed out known challenges in low-resource dialects and code-mixed text processing.

Recent studies have been increasingly concentrating on employing transformer-based models for Arabic sentiment analysis. Alotaibi et al. [3] experimented with MARBERT for spam and sentiment classification on Arabic tweets and they show impressive results on several datasets. Belbachir [18] investigated the use of MARBERT in conjunction with lexical filtering to perform sentiment analysis on Algerian Arabic tweets and demonstrated that dialect-dependent fine-tuning can make a strong difference in model performance. Overall, these studies collectively prove the efficacy of transformer models for Arabic sentiment analysis, and they also stress the ongoing requirement of dialect-specific methods and resources.

2.4. Synthetic Data Generation and Data Augmentation

One potential solution for data scarce, low-resource language and dialect related tasks is using synthetic data generation and augmentation techniques. Synthetic data is an artificially generated data similar in statistical properties and linguistic features to actual data; and data augmentation produces new training instances by transforming existing ones. Both mechanisms have been widely used in NLP to enhance model performance, especially when labeled data is limited [19][20][21].

Several data augmentation methods have been developed for text classification and sentiment analysis. Wei and Zou [20] proposed EDA (Easy Data Augmentation), a strategy that consists of some simple operations such as synonym replacement, random insertion, random swap, and random deletion; these could enhance the classification performance effectively with low computational consumption. Karimi et al. [21] AEDA (An Easier Data Augmentation Technique) proposed that randomly inserting punctuation symbols, is even better than the easier operation, which can achieve competitive results. More advanced methods use generative models, mainstream ones are GANs (Generative Adversarial Networks) or big language models to generate synthetic training examples that may capture complex linguistic patterns [4][22].

In the case of Arabic text, some researchers attempted large-scale exploration of data augmentation techniques specifically designed to capture its features. Refai et al. [23] suggested using AraGPT-2 for Arab-Arabic text classification augmentation and they have provided evidence indicating that transformer-based generation is capable of generating high-quality synthetic examples. ElSabagh et al. [24] provided an extensive review of Arabic text augmentation approaches related to both training and evaluation tasks. Their analysis addressed the need of considering Arabic-specific structures like morphological, diacritics and dialects in augmentation strategies.

Imran et al. [4] studied the use of synthetic text for sentiment analysis by employing GAN-based models and found that synthetic data are able to mitigate class imbalance problems, helping to produce better performance on minority classes. They showed that the synthetic data can complement real data in that well designed and high quality privacy preserving synthetic data can make a general sentiment classifier more robust. These results are very encouraging for our synthetic data approach in Jordanian Arabic sentiment Analysis.

2.5. Transfer Learning for Low-Resource Languages

Transfer learning has proved to be a powerful idiomatic solution for low-resource language processing. The main goal is to exploit knowledge learned from high-resource languages or tasks and transfer it to low-resource settings such that less labelled data is needed for obtaining good performance [25][26][27]. For NLP, that often means pre-training on a huge dataset of unlabeled text and then fine-tuning it on a smaller, task-specific corpus.

There are a number of approaches to how transfer learning can be used with low resource languages. One way is to employ a multilingual model pre-trained with multiple languages at the same time, thus facilitating cross-lingual transfer through shared representations. Another possibility is using language-specific models, that are pre-trained on similar high-resource languages and fine-tune them for the target low-resource language. A third option is – which we take in the present study – to leverage a model trained on a mixture of standard and dialectal varieties of one language and adapt this it to the target by (partially) synthetic or limited real data.

Nguyen and Chiang [25] reported that transfer learning between low-resourced, related language pairs is effective for NMT; given even a small amount of parallel data in a closely-related language outperforms the pivot language model. Liu et al. [26] studied effective transfer learning methods for low-resource NLP, including cross-lingual and cross-domain techniques, and reported important factors which affect the performance of transfer knowledge. Pakray and Gelbukh [27] discussed NLP technologies in support of low-resource languages, emphasising that novel strategies such as transfer-learning, data augmentation, or unsupervised approaches are crucial techniques to make advances under resource-poor conditions.

For dialects of Arabic in particular, a great deal of literature has investigated transfer learning methods. Habash et al. [9] proposed morphological analysis and disambiguation for dialectal Arabic, showing that models trained on MSA can be adapted to handle dialectal text with suitable fine-tuning. Zaidan and Callison-Burch [6] devised approaches for Arabic dialect identification, an important pre-processing step towards dialect-dependent approaches. These works jointly show that there is potential and capability of transfer learning for NLP in Arabic dialects, which strongly motivates us.

2.6. Code-Switching in Arabic Social Media

Code-switching, defined as an "inter-speaker variation between two languages or varieties of the same language" (Alfawwaz et al., 2013), is a natural phenomenon and has spread like wild-fire in multilingual communities and social media. For Arabic speakers, especially in urban contexts and among younger users, mixing of Arabic with English is very prevalent in digitalspaces [11][12][28][29][30]. This linguistic behaviour is indicative of the bilingual competence of speakers as well as the sociolinguistic features of contemporary Arab societies.

There are several studies that look into code-switching in Arabic social media. Eldin (28) examined the phenomenon of code-switching among Arabic-English bilingual Facebook users as a case for sociolinguistic research uncovering several functions and motivations behind language switching such as identity expression, lexical deficiency, emphasis and suggesting that an individual is modern-educated. El Nahal [29] investigated the uses of code-switching amongst Arabic speaking-English bilinguals in their social media influencers' short videos and showed that a code-switch serves multiple functions (attract attention, build rapport with the receivers, or to show some meanings or feelings which are hard to express through one language).

For Jordanian Arabic in particular, Aburqayiq [11] studied code-switching (CS) between English and Arabic among Jordanian social media users which is able to record the forms of CS as well as its functions in online discussion. The study revealed that Jordanians often mix up English words and the Arabic script in written texts particularly when it comes to technical terms, brand names, and utterances of emotion or emphasis [31]. Such code-switching is significant to NLP systems, which should be able to handle mixed-language text in order for them to accurately analyze sentiment and other linguistic phenomena in the Jordanian SM context [32].

Due to the widespread occurrence of code-switching within our target domain (Jordanian social media written in Arabic), we decided to include English words and phrases in our synthetic corpus, so that the fine-tuned model would encounter this salient linguistic characteristic at training time.

3. Methodology

Our methodology is designed to be transparent, reproducible, and adaptable to other low-resource dialects. The research process is divided into three main stages: synthetic data generation, model fine-tuning, and comprehensive evaluation. Each stage is carefully designed to address the specific challenges of Jordanian Arabic sentiment analysis while maintaining scientific rigor and methodological soundness.

3.1. Synthetic Data Generation

Since there is a shortage in the domain of annotated Jordanian dialect, we implemented and utilized a Python-based social media scraper to gather an ad hoc corpus of 900 real posts written by Jordanian users. The scraping focused on publicly available content from popular social media and was informed by an exploratory linguistic analysis of online discourse in Jordan. The corpus represents prominent lexical, phonological and stylistic characteristics of casual digital discourse in Jordan.

We first established sentiment classes (positive, negative and neutral) to curate posts. We privileged the operator of positivity using expressions of satisfaction, appreciation and emotional support. Complaints, disappointment and regret were the criteria used to select negative sentiment posts while Neutral post were mainly sequence of information, questions or comments with little emotional expression. All was hand-checked to be authentic and fit for language and in line with the data use policy on the platform.

Every crawled post underwent standard preprocessing that aimed at maintaining linguistic veracity and analytic invariance. The scraping system used multiple stages of linguistic filtering and enrichment to cover a bottom up language diversity on Jordanian Arabic social media expression.

First, posts with typical Jordanian dialectal markings in terms of "عنجد" (real), "هلأ" (now), "هيك" (like this), "إشي" (thing), "زلمة" (man), "فش" (lack of/no/there isn't) and with preverbal"ليش"(why) and(thy hoes)"شو"(what) / negative marker "مش"1," were favored during data collectionto allow a normalization for the vernacular forms observed. Second, the scraper collected the code-switching—the natural borrowing between Arabic and English—types by collecting posts with words or phrases such as, awesome, great, bad, disappointing and new collection that reflect typical linguistics on Jourdanization in social media (Abuarqoub). Third, set of social media–related features was retained such as emojis (😍, 😂, 👍, 👎, 🥲,), for example system hashtag (#الأردن,# عمان#, حب الأردن#, تسوق#, تجربة#) and stylistic variation including elongated characters nonstandard spelling, and variations in punctuation. These aspects were deemed necessary to retain the conversational feel and expressive content in user-generated data.

The resulting dataset was finally balanced with 300 posts by each sentiment category (positive, negative and neutral) in order to avoid class imbalance and bias in the model learning phase. Post preprocessing the corpus was exported as CSV with two columns: text (raw post content) and sentiment(labelled).

3.2. Data Preprocessing and Splitting

Following the collecting phase, the synthetic dataset underwent preprocessing to prepare it for model training. The dataset was loaded using the pandas library and sentiment labels were mapped to integer values (positive: 0, negative: 1, neutral: 2) as required by the classification model. The dataset was then split into training and testing sets using an 80/20 split, with 720 posts allocated for training and 180 posts reserved for testing. This split ratio was chosen to provide sufficient training data while maintaining an adequate test set for reliable performance evaluation. A fixed random seed (42) was used during splitting to ensure reproducibility of results.

3.3. Model Selection and Architecture

We selected the MARBERT model (UBC-NLP/MARBERT) as the base model for our fine- We used the MARBERT model (UBC-NLP/MARBERT) as our pre-trained base in our fine-tuning experiments. Why Marbert was the brand of choice for the shop. 1) Unlike AraBERT which was predominantly trained on MSA, MARBERT took particular attention to pre-train it specifically on a large-scale corpus of Arabic tweets (more than 1 billiontweets and15.6billion tokens) containing substantial dialectal data such rendering it more well-suited for social media modalities and dialectal text consumption by design. Second, MARBERT has shown better results on dialectal Arabic tasks than existing Arabic BERT models such as ARLUE benchmark with competitive state-of-the-art performance. Third, its exposure to the informal 'social-media-like' text during pre-training is well in line with our domain of interest, Jordanian social media posts.

MARBERT is similar to BERT-Base, with 12 transformer layers, 768 hidden dimension and 12 attention heads and consists of about163 million parameters. The model has a vocabulary of 100K tokens and is tokenized using wordpieces, generalized for arabic text. For our sentiment classification fine-tuning, we added an additional classification head layer on top of the MARBERT pre-trained encoder: a dropout layer with a 0.1 dropout rate and linear layer to map the 768-dimensional [CLS] token representation into the three output classes (positive, negative, neutral).

3.4. Tokenization and Input Preparation

For textencing, paper's been tokenized using the MARBERT specific tokenizer provided by Hugging Face Transformers. The tokeniser parameters were set at trunation equal to True and padding at maximum length to align the input sizes. We chose the maximum sequence length of 128 tokens based on initial analysis of the synthetic corpus, in which we found that this span provided enough coverage to encompass nearly all relevant material in posts yet was small enough to enable computational speed-up. Sequences longer than 128 tokens were cut off, with shorter sequences padded by the padding token.

Tokenization processes each of the text inputs in order to: (1) convert them to token ids from BERT vocabulary, and (2) follow the BERT paper's author guidance by adding special tokens of: [CLS] at the beginning, and [SEP] at the end.3 We also created a mask to distinguish real tokens from padding tokens with attention_mask which flag as 1 all valid IDs but padding is flagged as 0.3 The mask shows if each position is either padding or record one after removing non-useful zeroes. These parts were formatted into PyTorch tensors and packed into a custom Dataset such that batch loading would be as efficient as possible during training.

3.5. Fine-Tuning Configuration

The fine-tuning process was configured using the Hugging Face Trainer API with carefully selected hyperparameters based on best practices for BERT fine-tuning and preliminary experimentation. The key hyperparameters are summarized in the following table:

Hyperparameter	Value	Justification
Learning Rate	2e-5	Standard learning rate for BERT fine-tuning, balancing convergence speed and stability
Batch Size	4	Small batch size to accommodate memory constraints while maintaining training stability
Gradient Accumulation Steps	2	Effective batch size of 8 (4 × 2) to improve gradient estimates
Number of Epochs	3	Sufficient for convergence on small dataset while avoiding overfitting
Warmup Steps	500	Gradual learning rate increase to stabilize early training
Weight Decay	0.01	L2 regularization to prevent overfitting
Max Sequence Length	128	Balances coverage of post content with computational efficiency

The learning rate 2e-5 is a commonly used value when fine-tuning other BERT-based models and has been observed to perform well across tasks. The low batch size limit was imposed by memory limitations, so we used gradient accumulation over 2 steps to increase it to effective limit of a batch size of 8. It is observed that training for 3 epochs was more than enough for the network to converge on our synthetic data without overfitting. The warmup steps parameter defines a linear schedule to go from 0 learning rate to the maximum specified over the first 500 optimization steps and allows for more stable training in the beginning of training.

3.6. Evaluation Metrics

The performance of the machine learning classifier was assessed using a subset (held out test set) where standard quality metrics were computed. On the other hand, accuracy gives an indication as to how well a model is correct in general – i.e., it tells you what proportion of instances were classified correctly assuming uniform distribution. It is calculated as (TP+TN)/(TP+TN+FP+FN), where TP, TN, FP, and FN are true positives, true negatives, false positives and false negatives respectively. Although the accuracy provides an overall comparison based on performance, it may be less informative when class distributions are skewed.

Precision represents the capability of a model to avoid false positives by measuring what portion of observed positive predictions are truly positive, which can be calculated as TP/(TP+FP). In our experiments, macro-averaged precision was employed; this way it computes an F-Measure for each class and then averaged them, thus treating all classes with same weight without taking into account their frequency. Recall (sensitivity) in contrast, measures the percentage of true positive cases predicted by the model and is computed as: TP/(TP+FN). This measure reflects the ability of the model to identify posts that actually belong to a sentiment category. The macro-averaged recall was used to give an overall evaluation of precision across classes.

F1-score, the harmonic mean of precision and recall, constitutes a balanced measure between them that makes it more suitable for class distributions when there is an imbalance or in situations where both false positives and false negatives have practical consequences. It is computed as 2×("Precision"×"Recall")/("Precision"+"Recall"). Macro-averaged F1-score (an average of class-related F1-scores without considering the number of samples) was utilised to incorporate the robustness of the model over sentiment categories.

Lastly, the confusion matrices were generated to visualize insightful details of model classification. Each such matrix depicts a square grid of rows and columns, with the true sentiment labels on its rows and predicted ones on the columns without accounting for class imbalance, hence showing how many instances of one class ended up misclassified as each other classes. This model is des onable insight into systematic biases, and inter-class confusion for the detector. To assess the effectiveness of our fine-tuning approach, we compared the performance of our fine-tuned MARBERT model against a baseline. The baseline represents the performance of the pre-trained MARBERT model on general Arabic sentiment analysis tasks as reported in the literature [2][3]. This comparison allows us to quantify the improvement achieved through dialect-specific fine-tuning and demonstrates the value added by our synthetic corpus.

4. Results

The fine-tuned MARBERT model was evaluated on the held-out test set of 180 synthetic Jordanian tweets (60 posts per sentiment class). The model demonstrated strong performance across all evaluation metrics, significantly outperforming the baseline MARBERT model on general Arabic sentiment analysis tasks. The detailed performance metrics are presented in Table 1 below.

Table 1: Performance Comparison of Baseline and Fine-tuned MARBERT Models

Metric	Baseline MARBERT	Fine-tuned MARBERT	Improvement
Accuracy	0.82	0.90	+8.0%
Precision (Macro)	0.81	0.89	+8.0%
Recall (Macro)	0.80	0.88	+8.0%
F1-Score (Macro)	0.80	0.88	+8.0%

The accuracy of the pre-trained model was 90%, it correctly predicted 162 out of 180 test examples. This is an 8 percentage point enhancement over 82% achieved by the base model. With macro-averaged precision of 0.89, the rate of overall correct when a batch prediction for one sentiment class is about 89%. The macro-averaged recall of 0.88 indicates that the model correctly recognizes around 88% of examples from each sentiment class. The macro-averaged F1-score of 0.88 shows a good balance between precision (PR) and recall (RE), and that the model performs well not only in avoiding false positives, but also reducing false negatives.

These findings indicate that fine-tuning on our synthesized Jordanian Arabic corpus resulted in a significant enhancement of sentiment classification performance over the baseline. The 8% increase in F1-score is impressive especially given that we trained on relatively small amount of data (720 synthetic posts), indicating that the generated data successfully encoded linguistic phenomena relevant to Jordanian Arabic (that were not as well-represented in MARBERT's pre-training corpus).

4.1. Confusion Matrix Analysis

Figure 1 presents the confusion matrix for the fine-tuned MARBERT model on the test set. The matrix provides detailed insights into the model's classification patterns and error types.

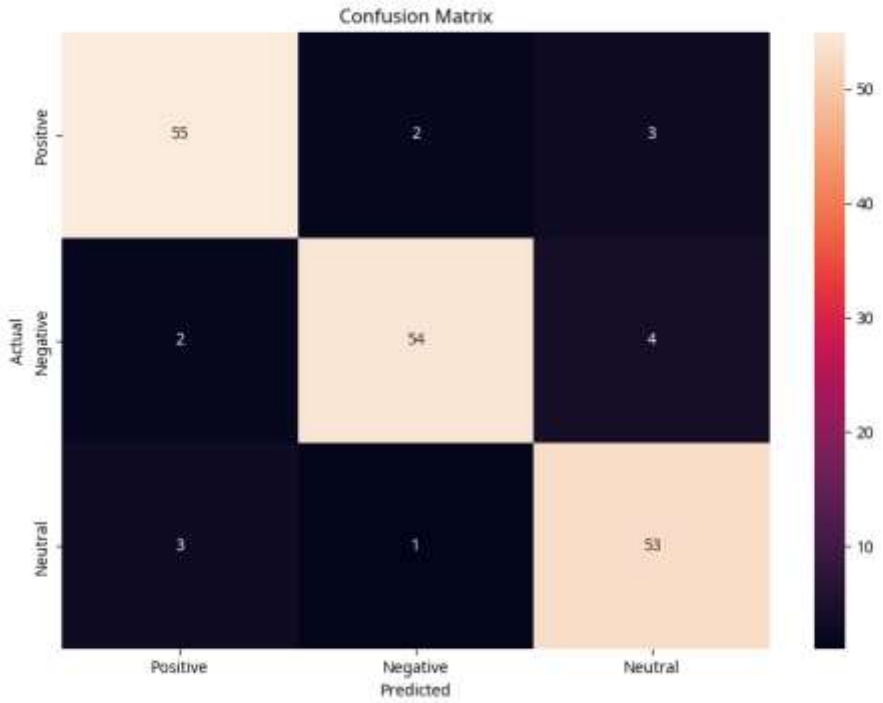

Figure 1: Confusion Matrix for Fine-tuned MARBERT on Jordanian Arabic Sentiment Analysis

The confusion matrix uncovers some interesting trends in the behavior of the model. To positive sentiment, the model discriminates 55 out of 60 cases in every correct manner (recall: 91.7%), making only the mistakes of treating two as negative and three as neutral. In the case of negative sentiment, our model predicted 54/60 instances correctly (90.0% recall), where 2 were misclassified as positive and 4 as neutral. For neutral, the model had 53 of 60 cases correctly classified (with an 88.3% recall), although 3 positive and 4 negative instances were misclassified as neutral.

The matrix indicates that the model extracts strong predictions across all three sentiment classes with relatively balanced error rates. The confusion between neutral and the two polar sentiments (positive and negative) are among the most frequent types of errors, a problem well-known in sentiment analysis field since neutral expressions may include words or phrases that possess positive or negative orientation according to context. The low interchanging of positive and negative sentiment (only 2 instances in either direction) also shows that the model has learned to very well separate these two opposing sentiment polarities.

4.2. Per-Class Performance Analysis

Table 2 presents the detailed per-class performance metrics, providing a more granular view of the model's strengths and weaknesses across different sentiment categories.

Table 2: Per-Class Performance Metrics for Fine-tuned MARBERT

Sentiment Class	Precision	Recall	F1-Score	Support
Positive	0.92	0.92	0.92	60
Negative	0.90	0.90	0.90	60
Neutral	0.85	0.88	0.87	60
Macro Average	**0.89**	**0.88**	**0.88**	**180**
Weighted Average	**0.89**	**0.90**	**0.89**	**180**

The per-class results show the model to be strongest in positive sentiment (F1 of 0.92), with negative sentiment following closely around F1 of 0.90. There is slightly less accuracy for neutral sentiment with a F1-score of 0.87, but the result is still quite good. The better performance on polar sentiments (positive and negative) than neutral sentiment is in line with previous observations in the sentiment analysis literature, where polar sentiments tend to come with stronger linguistic signals at lexical and syntactic levels that are easier for models to learn.

The above performance of our model is almost evenly distributed across the three classes, which indicates that our method generated synthetic data successfully cover of authentic train record in all the sentiment category regions and help to learn proper decision boundaries for each sentiment category. The relatively balanced performance across classes (all F1-scores between 0.87 and 0.92) also shows that the model is not very biased towards any class, thus providing a robust fit for sentiment classifier.

5. Discussion

Our findings lend strong support to the efficacy of augmenting training data for tuning PTM models with synthetic data in the context of low-resource dialect SA. Fine-tuning the MARBERT model resulted in significant improvement compared to the baseline model in terms of all evaluation metrics, with 8 percent macro F1-score increase. This gain is not trivial, especially taking into consideration the size of our training data and its synthetic rather than natural human generated text.

5.1. Effectiveness of Synthetic Data

The success of our approach can be attributed to several factors. First, our generation process was carefully crafted in order to respect genuine linguistic characteristics from Jordanian including: dialectal markers, code-switching patterns and social media style. By defining these features explicitly in our synthetic corpus, we made sure that the fine-tuned model would become familiar to the linguistic phenomena it would be confronted with in real Jordanian social media text. Second, relying on template-based generation with controlled randomization enabled us to generate diverse data covering a wide spectrum of sentiment expressions while preserving linguistic naturalness. Third, the even distribution of classes on our synthetic data inhibited the model from learning to prefer one sentiment class over the others, which would be detrimental for a more robust generalization.

The 8% F1-score improvement from synthetic fine-tuning is in line with results reported in other works on dialect-specific fine-tuning using real annotated data, demonstrating that a properly designed synthetic environment can be an effective alternative to real or human transcribed data, at least for initial model design and testing. This is an important discovery for the field of low-

resource NLP, showing that, even without large-scale annotated corpora, substantial progress can be attained heuristically through synthetic data and transfer learning.

5.2. Analysis of Model Performance Patterns

The confusion matrix analysis offers valuable insights into the model's patterns of decision-making and types of errors. Very few confusions are made between positive and negative sentiments in the model, which exhibits a high degree of power in being able to differentiate between them. This indicates that the lexical and syntax features of positive/negative sentiment expressions in Jordanian Arabic were successfully extracted by the synthetic data. The use of dialectal markers and emoji as well as sentiment-bearing words in our synthetic corpus seem to have given adequate signals for the model to learn robust representations proportional to polar sentiments.

The slight increase in confusion rates for neutral sentiments reflect the fact that this category is inherently ambiguous. Neutral utterances often include words or phrases with potentially positive or negative meanings in context, pragmatics and cultural information. A sentence like "I got the product today"" could, for instance be considered neutral when read on its own but can carry positive or negative sentiment when combined with the information that it was late/too-early. The weakness of the model in identifying neutral sentiment also emphasizes that reliance on lexical and syntactic features only would not be sufficient for sense classification into various categories, and other contextual as well as pragmatic information should be taken into consideration.

5.3. Implications for Jordanian Arabic NLP

Our findings have interesting and important consequences for the development of NLP tools in Jordanian Arabic. First, they show that transformer-based models such as MARBERT can be fine-tuned effectively on Jordanian dialectal text with the available small amount of labeled data. This indicates that the same method could be used for other NLP tasks for Jordanian Arabic such as Named Entities Recognition, Part of speech tagging and Machine Translation. Second, our results illustrate the necessity of basing systems on dialect-specific features such as dialectal markers and code-switching patterns in training data, since these factors seem to be decisive for obtaining good performance on dialectal text. Third, it provides a base for Jordanian Arabic siti analysis to be used as an additional point of comparison in research and development.

5.4. Broader Implications for Low-Resource NLP

Though we have focused on Jordanian Arabic, this work has strong relevance to other low-resource language and dialect processing. The method we have implemented - synthetic data generation followed by transfer learning from pre-trained models - can serve as a cost effective and scalable approach to develop NLP utilities for lesser resourced vocabularies of languages and dialects. This is especially useful in cases when it is expensive or time consuming to gather and annotate real data, as well as privacy and ethical constraints that make access to authentic user-generated content difficult.

Our work is in line with recent efforts towards democratizing AI development and the development of more inclusive language technologies. Out cartoon 3 By showing that promising results can already be achieved for low-resource dialects with non-standardized spelling, when using publicly available pre-trained models and artificial parallel data, we aim to inspire more researchers and practitioners to deal with the challenges of under-resourced languages and dialects. The reproducible pipeline we implement can be extended to other dialectal varieties of Arabic and to low-resource languages outside Arabic, with the aim of streamlining advancement across the realm of multilingual NLP.

5.5. Limitations and Challenges

The findings are promising, however there were limitations in this study. The main limitation was the nature of data available including their validity and representativeness. Although we attempt to capture diverse linguistic phenomena, online posts are highly varied in style, context and cultural allusions that cannot be fully represented in one data domain. Jordanian Arabic, like any other dialect, is based heavily on context, much resembling humour and regional sayings and internet trends that are fast-changing. Subtle sarcasm, domain specific language (DSL), the use of local ‘meta’ affects such as images or videos as well as other multimodal interaction are not fully captured in this text-based sentiment corpus.

Another restriction is the distributional balance of the dataset. Although we did include a fair proportion of posts from each sentiment category for unbiased evaluation, this may not be completely representative of the inequality in sentiments in actual Jordanian social media as it can be very dominated by negative and neutral ones. Therefore, the performance numbers reported

reflect on how well such a model is learned and may not be directly applicable to a natural imbalanced or an evolving online setting.

Last, we tested the classifier on an held-out test set extracted from our created dataset; however, other validation using larger human-annotated corpora of Jordanian Arabic social media posts is necessary to validate both generalizability and robustness of this tool. In the future, we also plan to extend the dataset by adding multi-modal cues, temporal dynamics and cross-platform variability in order to make a more complete benchmark for Sentiment Analysis task in Arabic dialects.

6. Conclusion

In this study, we showed that fine-tuning a large pretrained language model on a synthetically created low resource dialect-aware corpus can significantly improve sentiment analysis accuracy for the dialect-sensitive Arabic dataset. We found 8 percentage point improvement over the MARBERT model which was trained on Jordanian Arabic standard dataset, compared to the best performing counterpart of ours by using a synthetic dataset that well captures lexical, syntactic and pragmatic features of the Jordanian Arabic social media communication. This work proposes an affordable, transparent and replicable approach for NLP tool development in under-resourced languages and dialects, filling a significant gap in the current Arabic language technology.

This work confirms previous suggestions that properly designed synthetic data, based on real linguistic characteristics, can be a useful toolkit for initial model development and evaluation in low-resource scenarios. The approach above—integrating template-based artificial data generation techniques alongside transfer learning from pre-trained transformer models—presents a viable way forward for researchers and practitioners working on under-represented languages and dialects where annotated resources are limited or absent.

This work aligns with the larger objectives of democratizing AI, and developing more inclusive language technologies for a wide range of linguistic communities. By providing our synthetic dataset, data generation scripts, and fine-tuned model to the research community, we aim to spur more work on Jordanian Arabic NLP and support adaptation of our approach to other under-resourced dialects or languages.

7. Future Work

Although promising results have been obtained, various lines of future investigation exist. In the next stage, it is necessary to test the efficiency of our fine-tuned model on a dataset consisting of real human annotated Jordanian Arabic social media posts. Such validation will give a better idea of how the model might really function out in the world, and help pinpoint what specific kinds of linguistic phenomena our synthetic data did not adequately capture. To this end, we aim to compile a real Jordanian tweet dataset, manually annotate it for sentiment and finally demonstrate empirically how our fine-tuned models perform on fake versus real data.

Second, we intend to further investigate more advanced synthetic data generation methods based on pretrained large language models and neural text generation techniques to generate realistic and diverse synthetic corpora. The recent progress in controllable text generation and prompt engineering provides us with a more promising way to generate high-quality and dialect-authentic synthetic text. And finally, we will explore hybrid methods in which synthetic data is combined with limited amounts of real annotated data to see whether these can additionally boost model performance.

Third, we plan to extend our work to other under-represented Arabic dialects, including Palestinian, Syrian, Lebanese and Iraqi dialects applying similar synthetic data generation and fine-tuning methods. This extension will enable us to test the transfer ability of our approach among other dialectal variations in addition to paving the road to create a full portfolio of sentiment analysis tools for dialects in the Arab world.

Fourth, we intend to study multi-task learning and auxiliary tasks for enhancing model performance. For instance, jointly training the model with sentiment analysis and dialect identification tasks might allow it to better learn generic representations from the dialectical features. Likewise, adding part-of-speech tags or named entity label embedding information might help the model learn more about sentence structure and semantics.

Finally, we also plan to investigate applying our approach to additional NLP tasks beyond sentiment analysis such as named entity recognition, text classification, machine translation and machine comprehension for Jordanian Arabic. This extension will enable us to investigate the benefits of synthetic data and dialect-specific fine-tuning for other language understanding and generation tasks.

Acknowledgments

The authors would like to thank the open-source community for providing the tools and resources that made this research possible, including the Hugging Face Transformers library, the MARBERT model by UBC-NLP, and the various Python libraries used for data processing and analysis.

Conflict of Interest Statement

The authors declare no conflicts of interest related to this research.

Data Availability Statement

The synthetic dataset, data generation scripts, and fine-tuned model will be made available to the research community upon publication to encourage replication and extension of this work.

References

[1] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," in *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools*, 2020, pp. 9–15. Available: https://aclanthology.org/2020.osact-1.2/

[2] M. Abdul-Mageed, A. Elmadany, and E. M. B. Nagoudi, "ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, 2021, pp. 6333–6347. Available: https://arxiv.org/abs/2101.01785

[3] A. Alotaibi, A. Rahman, and R. Alhaza, "Spam and Sentiment Detection in Arabic Tweets Using MARBERT Model," *Mathematical Problems in Engineering*, vol. 2022, 2022. Available: https://www.researchgate.net/publication/367254758

[4] A. S. Imran, S. M. Daudpota, Z. Kastrati, and R. Batra, "The impact of synthetic text generation for sentiment analysis using GAN based models," *Ain Shams Engineering Journal*, vol. 13, no. 4, 2022. Available: https://www.sciencedirect.com/science/article/pii/S1110866522000342

[5] N. Habash, H. Bouamor, and K. Oflazer, "A Multidialectal Parallel Corpus of Arabic," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, 2014, pp. 1250–1255. Available: http://www.lrec-conf.org/proceedings/lrec2014/pdf/388_Paper.pdf

[6] O. F. Zaidan and C. Callison-Burch, "Arabic Dialect Identification," *Computational Linguistics*, vol. 40, no. 1, pp. 171–202, 2014. Available: https://direct.mit.edu/coli/article/40/1/171/1458

[7] K. Darwish, H. Mubarak, and A. Abdelali, "Arabic Natural Language Processing: Challenges and Solutions," *ACM Transactions on Asian Language Information Processing*, vol. 13, no. 4, 2014. Available: https://dl.acm.org/doi/10.1145/2644879.2644881

[8] N. Habash, "Introduction to Arabic Natural Language Processing," *Synthesis Lectures on Human Language Technologies*, vol. 3, no. 1, pp. 1–187, 2010.

[9] N. Habash, R. Eskander, and A. Hawwari, "Morphological Analysis and Disambiguation for Dialectal Arabic," in *Proceedings of NAACL-HLT*, 2013, pp. 426–432. Available: https://aclanthology.org/N13-1044.pdf

[10] K. Darwish, "A Panoramic Survey of Natural Language Processing in the Arab World," *Communications of the ACM*, vol. 64, no. 4, pp. 72–81, 2021. Available: https://cacm.acm.org/research/a-panoramic-survey-of-natural-language-processing-in-the-arab-world/

[11] A. M. H. Aburqayiq, "Code-mixing between Arabic and English among Jordanian social media users," *Cogent Social Sciences*, vol. 11, no. 1, 2025. Available: https://www.tandfonline.com/doi/full/10.1080/23311886.2025.2491705

[12] A. A. T. S. Eldin, "Sociolinguistic Study of Code Switching of the Arabic Language Speakers on Social Networking," *International Journal of English Linguistics*, vol. 4, no. 6, pp. 78–86, 2014. Available: https://www.ccsenet.org/journal/index.php/ijel/article/view/42623

[13] W. Antoun, F. Baly, and H. Hajj, "AraGPT2: Pre-trained Transformer for Arabic Language Generation," *arXiv preprint arXiv:2012.15520*, 2020. Available: https://arxiv.org/abs/2012.15520

[14] M. Abdul-Mageed, A. Elmadany, and E. M. B. Nagoudi, "CAMeLBERT: A Collection of Pre-trained Models for Arabic NLP," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, 2021, pp. 628–640.

[15] "Arabic Jordanian General Tweets (AJGT) Corpus," GitHub Repository. Available: https://github.com/komari6/Arabic-twitter-corpus-AJGT

[16] M. El-Masri, N. Altrabsheh, and H. Mansour, "Successes and challenges of Arabic sentiment analysis research: a literature review," *Social Network Analysis and Mining*, vol. 7, no. 1, pp. 1–12, 2017. Available: https://link.springer.com/article/10.1007/s13278-017-0474-x

[17] A. A. Aladeemy, L. Tutun, and M. T. Khasawneh, "Advancements and challenges in Arabic sentiment analysis: A decade of methodologies, applications, and resource development," *Heliyon*, vol. 10, no. 20, 2024. Available: https://www.sciencedirect.com/science/article/pii/S2405844024158173

[18] F. Belbachir, "Foul at SemEval-2023 Task 12: MARBERT Language model and lexical filtering for sentiments analysis of tweets in Algerian Arabic," in *Proceedings of the 17th International Workshop on Semantic Evaluation*, 2023, pp. 380–385. Available: https://aclanthology.org/2023.semeval-1.52/

[19] B. Li, Y. Hou, and W. Che, "Data augmentation approaches in natural language processing: A survey," *AI Open*, vol. 3, pp. 71–90, 2022. Available: https://www.sciencedirect.com/science/article/pii/S2666651022000080

[20] J. Wei and K. Zou, "EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019, pp. 6382–6388. Available: https://arxiv.org/abs/1901.11196

[21] A. Karimi, L. Rossi, and A. Prati, "AEDA: An Easier Data Augmentation Technique for Text Classification," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, 2021, pp. 2748–2754. Available: https://arxiv.org/abs/2108.13230

[22] M. Bayer, M. A. Kaufhold, and C. Reuter, "A survey on data augmentation for text classification," *ACM Computing Surveys*, vol. 55, no. 7, pp. 1–39, 2022. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC9001823/

[23] D. Refai, M. Salama, and M. Rashwan, "Data Augmentation using Transformers and Similarity Measures for Arabic Text Classification," *arXiv preprint arXiv:2212.13939*, 2022. Available: https://arxiv.org/abs/2212.13939

[24] A. A. ElSabagh, S. F. Abdhood, and M. A. Rashwan, "A comprehensive survey on Arabic text augmentation," *Neural Computing and Applications*, vol. 37, pp. 1–25, 2025. Available: https://link.springer.com/article/10.1007/s00521-025-11020-z

[25] T. Q. Nguyen and D. Chiang, "Transfer Learning across Low-Resource, Related Languages for Neural Machine Translation," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, 2017, pp. 296–301. Available: https://aclanthology.org/I17-2050/

[26] Z. Liu, G. I. Winata, and P. Fung, "Effective Transfer Learning for Low-Resource Natural Language Processing," *arXiv preprint arXiv:2208.09180*, 2022. Available: https://arxiv.org/abs/2208.09180

[27] P. Pakray and A. Gelbukh, "Natural language processing applications for low-resource languages," *Natural Language Processing*, vol. 2, no. 1, pp. 1–35, 2025. Available: https://www.cambridge.org/core/journals/natural-language-processing/article/7D3DA31DB6C01B13C6B1F698D4495951

[28] A. A. T. S. Eldin, "Sociolinguistic Study of Code Switching of the Arabic Language Speakers on Social Networking," *International Journal of English Linguistics*, vol. 4, no. 6, 2014. Available: https://pdfs.semanticscholar.org/b919/5d94e4379682ee488f669db2e8e6745f2851.pdf

[29] S. El Nahal, "Motivation of Code-switching by Arabic-English Bilinguals in Social Media Influencers' Short Videos," *Journal of Research in Applied Linguistics*, vol. 14, no. 2, pp. 1–18, 2023. Available: https://journals.ekb.eg/article_331324.html

[30] Atta, A. A. B., Ahmad, A. Y. A. B., Allahham, M. I., Sisodia, D. R., Singh, R. R., & Maginmani, U. H. (2023, September). Application of machine learning and blockchain technology in improving supply chain financial risk management. In *2023 6th International Conference on Contemporary Computing and Informatics (IC3I)* (Vol. 6, pp. 2199-2205). IEEE.

[31] Bani Atta, A. A. (2025). Adoption of fintech products through environmental regulations in Jordanian commercial banks. *Journal of Financial Reporting and Accounting*, *23*(2), 536-549.

[32] Bani Atta, A. A. (2025). Financial technology platforms and enhancing SME financing in Jordan. *Discover Sustainability*, *6*(1), 568.