Al-Nahrain Journal of Science
Al-Nahrain J. Sc., Vol. 29, Issue 2, pp. 120 – 133, June, 2026.

Proposing a Hybrid Model to Enhance the Prediction of Arabic Fake News: A Comparative Analysis

Khaldoon H. Al-Hussayni[1,*], Amina A. Dawood[2], Ahmed Al-Azawei[3]

[1] Cybersecurity Department, Information Technology College, University of Babylon, Babil, Iraq.
[2] Computer Centre, University of Babylon, Babil, Iraq.
[3] Software Department, Information Technology College, University of Babylon, Babil, Iraq.

Article's Information

Received: 04.07.2025
Accepted: 30.01.2026
Published: 15.06.2026

Keywords:
Fake news detection
Arabic language
Deep learning
CNN
BiLSTM
GRU
AraBERT

Abstract

The spread of fake news within social media networks in recent years is seen as a troublesome element in the flow of public information. Malicious actors, such as some media professionals, use fabricated stories to influence vulnerable social groups. In this paper, a hybrid model is proposed to identify disinformation in Arabic language material. A pre-training algorithmic framework for detection and categorization of fake news is developed using deep learning approaches based on the Arabic Fake News Dataset (AFND) with approximately 374,000 labeled articles (80% training and 20% testing). The proposed model was compared with baseline architectures, such as CNN-BiLSTM, CNN-BiGRU, standalone recurrent networks, and pretrained language models, such as AraBERT, MARBERTv2, and AraELECTRA. The results show that the hybrid AraBERT-BiLSTM model outperforms all baseline architectures with 98.16%, 98.16%, 98.16%, and 96.58% accuracy, precision, recall, and F1-score, respectively. This is because AraBERT offers rich contextual embeddings that extract the deep semantic and syntactic nuances of Arabic text. Additionally, the model interpretability was improved by using Local Interpretable Model-agnostic Explanations (LIME) to identify the important linguistic features driving predictions.

http://doi.org/10.22401/ANJS.29.2.10
*Corresponding author: alhussayni@uobabylon.edu.iq

1. Introduction

Social media networks (SMNs), such as Twitter, Facebook, and LinkedIn, are used in everyday life for education, entertainment, social interactions, and news tracking [1]. Many SMNs are used as a major source of rich information and as a means for news transmission and propagation. Ensuring the veracity of the news is an important factor in the credibility of a range of news sources [1]. On the other hand, SMNs allow users to post a variety of information, which may result in an increase in the quantity of web content. On the other hand, false news or misinformation may be spread either intentionally or unintentionally as no verification on this information is provided [3]. Misinformation may deceive people, mislead information, or cause harm. Hence, fake news can affect different life domains, such as the political and financial sectors [4]. Discriminating between true and fake news can reduce their negative impacts. Most fake news detection studies focused on the English content. Arabic fake news detection remains rare. This is because of the challenges associated with Arabic word recognition and language processing [5]. Most models struggle with Arabic's morphological complexity, dialectal variation, and limited annotated resources [6]. Hybrid architecture has not been fully leveraged to capture both contextual and sequential dependencies. This would further validate the novelty and necessity of our proposed AraBERT–BiLSTM model [7]. In addition, most of the previous work focused on specific algorithms rather than data preprocessing. This study tackles some of the important challenges in the area of false Arabic news detection by dealing with specific problems in Arabic text processing. This is based on improving an algorithm for detecting and classifying false Arabic news. This was accomplished through deep learning

and pre-training evaluation with basic models (Convolutional Neural Networks or CNNs, Bidirectional Recurrent Neural Network or BiLSTMs and BiGRUs), advanced Arabic language models AraBERT, AraELECTRA, MARBERTv2, hybrid models combining techniques, such as Convolutional Neural Network-Bidirectional recurrent neural network (CNN-BiLSTM) CNN-BiGRU, and AraBERT-LSTM to identify which combination is the most effective in detecting and classifying fake news. The contributions of this work are threefold. First, it directly addresses the challenges of the Arabic language in terms of morphological complexity, multiple dialects, and the scarcity of available data. Second, the research proposes an adaptive hybrid model that integrates the individual strengths of each algorithm to achieve superior performance. Finally, it implements an interpretation technique called Local Interpretable Model-agnostic Explanations (LIME) to interpret the model's prediction. The remainder of this study is organized as follows. Related literature is reviewed in Section 2. The proposed methodology is presented in Section 3. Section 4 reports and discusses the research findings. Finally, Section 5 concludes the study and highlights possible future directions.

2. Theoretical Background

Fake news detection is a challenging text classification problem. This is a difficult task for the complex Arabic which has various dialects. This may interpret why most of the earlier literature focused on English datasets [8][9]. The spread of fake news in Arabic has the potential to affect public opinion drastically, destabilize politics and bottom out trust or society. The problem has been studied from different perspectives using conventional machine learning algorithms, deep learning, and advanced transformer models [10, 11]. Researchers tackled this problem with classic machine learning techniques, such as Support Vector Machines (SVM), Decision Trees (DT), Naïve Bayes (NB), and Random Forest (RF). The common practice was to pair these algorithms with handcrafted features, primarily using TF-IDF, word embeddings from Word2Vec, and GloVe [12]. In [13] a modified Random Forest with fuzzy logic was applied to achieve an overall 89.5% accuracy. Another analysis confirmed that SVM and Random Forest were strong performers, although they had limitations when texts are linguistically complex [11]. Deep learning flipped scripts by automatically learning text representations. The first entrants in Arabic fake news detection were the CNNs and RNNs, particularly Long Short-Term Memory (LSTM) and Bi-LSTM that managed to capture local and sequential patterns [14][15]. An interesting hybrid architecture using CNN in connection with LSTM was reported for their Arabic dataset, achieving an accuracy of 96%. Similarly, Bi-LSTM showed the way it captured sequence dependencies in text [16]. Transformers, BERT, RoBERTa, and AraBERT models have been commonly used for Arabic fake news detection. They can learn about contextual and semantic natures that previously models struggled with [11][16]. MARBERTv2 was presented in [11], a multilingual Arabic dialect speaking transformer, which achieved an F1 94.12%. In [17], AraBERT was applied to detect fake COVID-19 vaccine tweets with 95% accuracy. Some researchers have developed creative and mixed architectures. Hybrid models like CNN-LSTM are constructed aiming to leverage benefits of both worlds, feature extraction ability of CNN and sequential modelling capability of LSTM [13, 18]. Ensemble learning has been a widely used strategy which integrates predictions from different models to achieve more stable results [19, 20]. Bayesian neural networks for measuring prediction uncertainty have only recently started to be adopted in medical imaging to measure prediction uncertainty, providing better insight into how confident such models are [19]. Pretrained language models, such as AraBERT, MARBERTv2, and AraELECTRA have all been fine-tuned for fake news detection, showing remarkable results [17][21][22]. AraBERT, essentially BERT speaking Arabic, has become a popular tool. In [14], it was paired with CNN architectures in which it achieved an impressive accuracy across multiple datasets. The MARBERTv2's ability to handle multiple Arabic dialects has been particularly impressive in which it achieved 94.12% F1 score [11]. WaraBERT then combined word-level tokenization with AraBERT variants in which it performed well with Arabic fake news datasets [23]. Based on this discussion, the field requires larger, and more diverse Arabic datasets. It is important to determine how to make these models more interpretable. It might be worth exploring multitask learning to tackle related problems such as rumor detection and sentiment analysis simultaneously. Most studies focused on English; however, Arabic remains under exploration. Combined algorithms have not yet been fully leveraged, and models still struggle generalizing their findings. A state-of-the-art model for Arabic language processing is so-called ArabertV2. It was developed by the Arabic Language Technologies group at the American University of Beirut [24]. It adopted Google's BERT model to handle Arabic Natural Language Processing related tasks. This

model was trained on various Arabic corpora that comprise a mammoth dataset of 200 million sentences with 8.6 billion words totaling 77 GB [25] The model comes in two versions; a Basic version with 136 million parameters (453MB), and a large version with 371 million parameters (1.38GB). The model utilizes enhanced preprocessing features such as Farasa Segmenter to leverage the model understanding of complex Arabic linguistic structures. Furthermore, it derives a new vocabulary for better recognition of the nuances of Arabic language morphology and syntax. The ArabertV2 performs with significant efficiency in natural language processing (NLP) tasks, such as semantic analysis, Named Entity Recognition (NER), and question answering.

2.1. Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) are considered as a category of Deep Learning (DL), specially modeled for images and time-series [26]. CNNs demonstrate high performance in feature selection and pattern recognition tasks. This type of network conveys structural features, including hierarchical feature learning. Lower layers can extract simple features such as edges, while upper layers expand these features to more complex concepts such as faces or objects. Additionally, the local connectivity technique inherited in CNNs, where each layer connected to small region from the parent layer. This feature allows CNNs to draw their attention to local phenomenon. Furthermore, across multiple spatial locations, filters can shear their weights to benefit from the reduction of the model dimensionality and increase efficiency. The basic components of CNNs are Convolution layers, Activation functions, Pooling layers, fully connected layers (Dense), and optionally Dropout layers. The Convolution layers apply sliding kernels (filters) across the input stream; they can capture basic features such as edges, textures or basic shapes. Such layer's yield feature maps. Another component is the Activation function, which introduces non-linearity to the learning curve of the model without those, CNN models will behave as Linear Regression models. An example of the activation functions is the Rectified Linear Unit (ReLU) which is a simple mathematical formula to convert negative values into Zero, Max (0, x). Pooling layers can help reduce the special dimensionality of the feature maps produced from the convolution layers. In other words, they perform a down sampling operation to increase computations efficiency and decrease the chance of overfitting. Fully connected layers act as the traditional neural networks to flatten the output of the previous layers to perform the final predictions. Optionally, the Dropout layers randomly prevent a portion of the previous layer neurons from participating in the processes of the next layer, concluding better mitigation of overfitting. When all of these components work, they can produce efficient models that can provide high performance with optimized efficiency to complete sophisticated tasks, such as audio and video analysis, medical imaging, and NLP.

2.2. Long-Short-Term-Memory (LSTM)

Long Short-Term Memory (LSTM) was introduced by Sepp Hochreiter and Jurgen Schmidhuber in 1997 [27]. It is considered as a type of Recurrent Neural Networks (RNN). LSTM was proposed to model streams, such as sequences and time-series and mimic human brain in remembering long-term dependencies. LSTM outperforms RNNs by overcoming the problems of vanishing (when repeatedly multiplying weights <1) and exploding (when repeatedly multiplying weights >1) gradients through a technique called the cell state. The use of cell state enables LSTM to efficiently process long sequences and expand their usage into various domains [28]. The input to an LSTM cell at each time step t maintained using a Hidden state (H_t) for short-term dependency and Cell state (C_t) for long-term dependency. The principal components of LSTM are Gates. LSTM uses three types of gates (i.e., Forget gate, Input gate, and Output gate) as illustrated in Figure 1.

Al-Nahrain Journal of Science
Al-Nahrain J. Sc., Vol. 29, Issue 2, pp. 120 – 133, June, 2026.

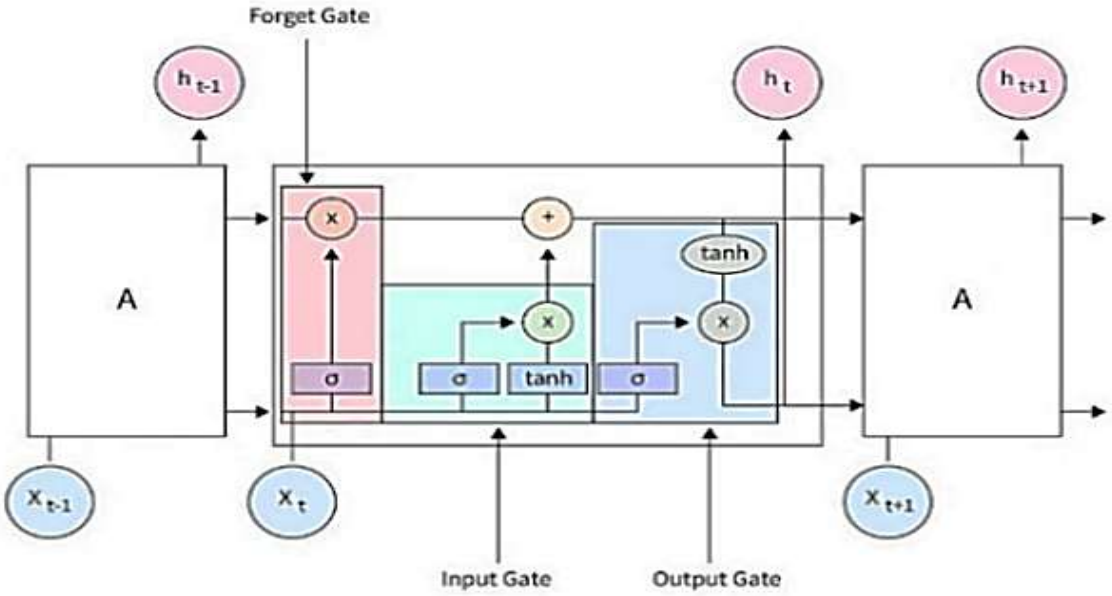

Figure 1: The LSTM architecture.

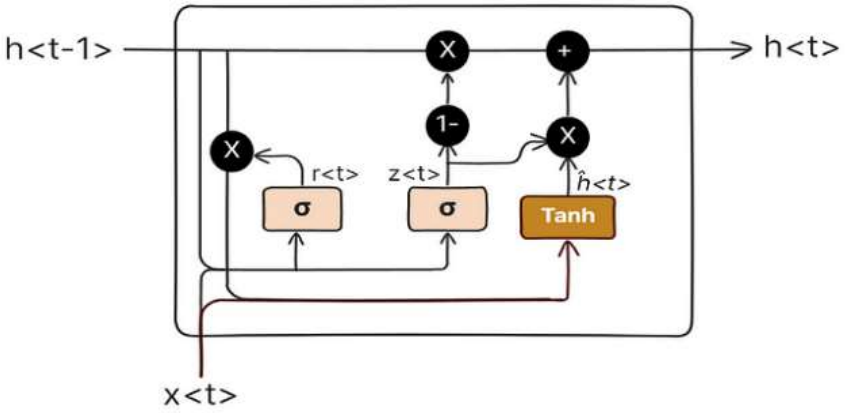

Figure 2: The GRU unit structure.

The Forget gate (f_t) decides which input from the hidden state H_t to discard (forget), using a Sigmoid function as shown in Equation 1. The second gate is the Input gate (i_t), which decides what new information should be injected into the cell. It also uses a Sigmoid activation function (see Equation 2). The last gate is the Output gate (o_t), which decides the information to keep from the cell state to pass to the output (see Equation 3).

$$f_t = \sigma\left(w_f[h_{t-1}.x_t] + b_f\right) \quad \ldots(1)$$
$$i_t = \sigma(w_i[h_{t-1}.x_t] + b_i) \quad \ldots(2)$$
$$o_t = \sigma(w_o[h_{t-1}.x_t] + b_o) \quad \ldots(3)$$

In addition to those three gates, LSTM has two cells: the Candidate cell ($\tilde{C}_t$) and the Update cell (C_t). The Candidate cell provides suggested update temporarily calculated at each time step based on the current input and the previous hidden state as shown in Equation 4. The Forget gate and the Input gate are merged at the Update cell (Equation 5) to update the long-term memory.

$$\tilde{C}_t = \tanh(W_C.[h_{t-1} + x_t] + b_C) \quad \ldots(4)$$
$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad \ldots(5)$$

To summarize the flow of LSTM cell at each time step (t), the cell obtains the current input (x_t) and the previous hidden state (h_{t-1}). Then, using the Update cell (C_t), it updates its cell state from the Forget and the Input gate. Finally, it produces the current

hidden state (h_t)). The structure of LSTM helps with versatile usage across AI applications, such as NLP.

2.3. Gated Recurrent Unit (GRU)

Gated Recurrent Units (GRUs) are considered as a simplified version of LSTMs, in the way that they merge some gates and the segregation of the independent cell state. GRUs still have the same strength of LSTM structure, but with a smaller number of parameters. The GRU cell structure comprises two gates as illustrated in Figure 2. the Update gate (Z_t), and the Reset gate (r_t), beside two states which are the Candidate hidden state ($h_t^{\sim}$) and the Final hidden state (ht). Noticeably, the absence of a separate Cell state (C_t) is because the LSTM cell derived by the Final hidden state (ht). The workflow of a GRU cell is summarized at each time step (t). First, the Update gate (z_t) controls the amount of information to keep from the previous Final hidden sate (h_{t-1}). The Update gate uses the Sigmoid activation function to maintain the elimination process, if (z_t) is almost 1, then keep the old state, otherwise use the Candidate state (Equation 6). On the other hand, the Reset gate (r_t) controls how much to forget from the past state. This gate also uses a Sigmoid function for activation. If r_t near 0, then forget the past state, otherwise use the full memory (see Equation 7).

The Candidate hidden state ($h_t^{\sim}$) uses the Reset gate rt to suggest a new hidden state (as in LSTM), using the hyperbolic tan activation function (Equation 8). Finally, the Final hidden state (ht) merges with the past Hidden state (h_{t-1}) and the Candidate state utilizing the Update gate (z_t) as in Equation 9. This state decides whether to keep the past state (when Z_t = 0) or otherwise use the suggested Candidate state [19].

$$Z_t = 2(W_z.[h_{t-1}, X_t] + b_z) \quad \dots(6)$$
$$r_t = 2(W_r.[h_{t-1}, X_t] + b_r) \quad \dots(7)$$
$$h_t^{\sim} = tanh(W_h.[r_t.h_{t-1}, X_t] + b_h) \quad \dots(8)$$
$$h_t = (1 - z_t).h_{t-1} + z_t.h_t^{\sim}) \quad \dots(9)$$

Table 1 shows a summarized comparison between LSTM and GRU, which illustrates the key differences between the two techniques.

2.4. Hybrid models

The idea of hybrid modes is to combine two or more models to come up with a new model (hybrid). This strategy is beneficial in many ways, such as overcoming the weaknesses of individual model. Another usage is to benefit from the output of one model as input to the other model for generalization purposes. This research uses the sophisticated pre-trained model AraBERT for its proven capabilities to capture Arabic language complex contextual nuances [29]. The proposed model combines AraBERT as language contextual transformer with LSTM for processing the sequential information output. The structure of the proposed model utilizes an initial preprocessing layer for the input textual data. A CNN layer is to extract the features map. The features map serves as an input to LSTM (or GRU) layer to extract the words dependencies and semantic interpretation. This approach contributes to the efforts of understanding Arabic linguistic structures, especially in the field of news contents analysis. The model demonstrates the potential strong implication of integrating neural networks (NNs) to predict and classify complex linguistic context with high efficiency.

Table 1: A summarized comparison between LSTM and GRU

Feature	GRU	LSTM
Gates	2 (update, reset)	3 (input, forget, output)
Cell state	uses hidden state only	separate cell state Ct
Complexity	Lower	Higher
Performance	Often similar in practice	Slightly better on complex tasks

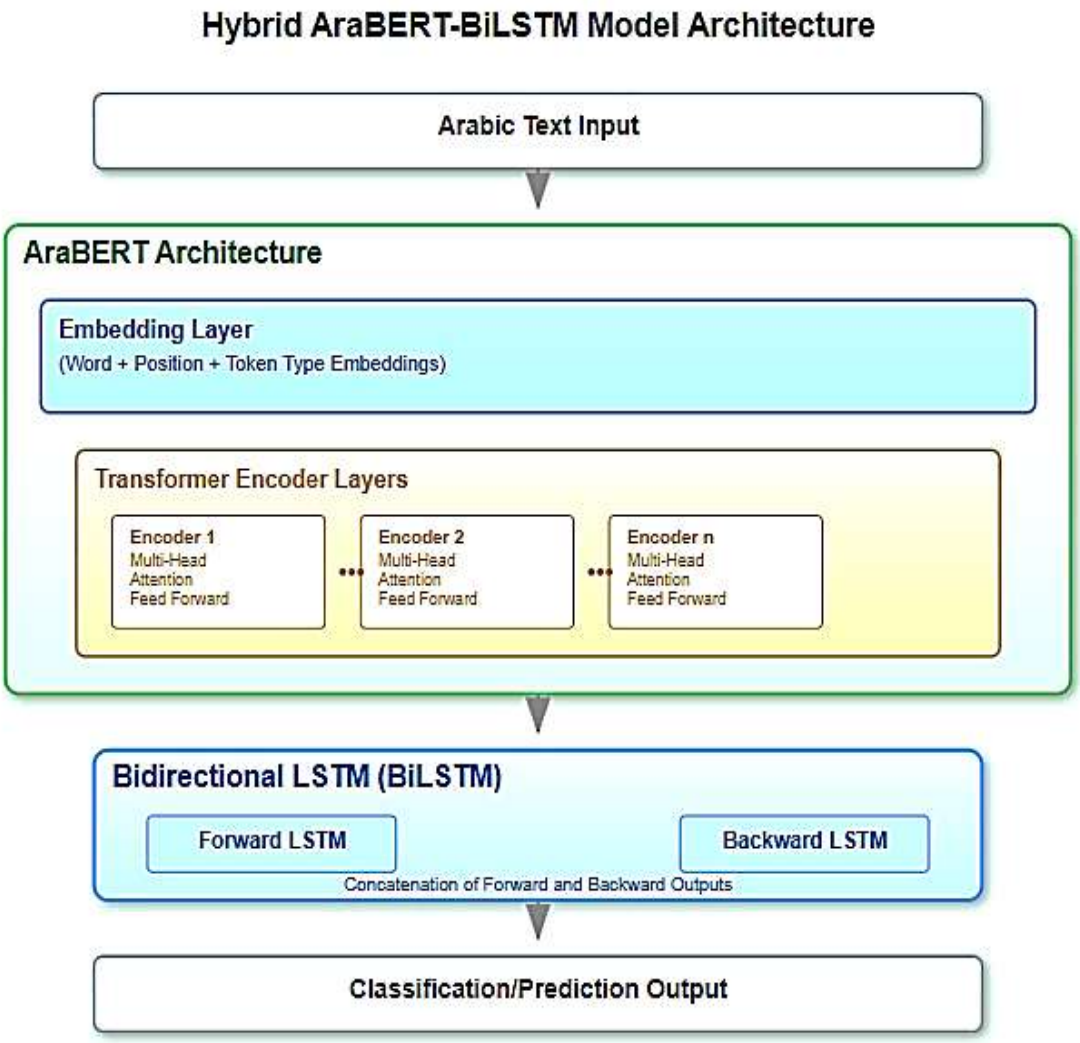

Figure 3: The proposed ArabBERT-BiLSTM model structure.

2.5. Local Interpretable Model Agnostic Explanations (LIME)

Local Interpretable Model-agnostic Explanations (LIME) works by perturbing the input data and observing how the model predictions change. For X_t classification tasks, LIME generates variations in the input text by removing or altering words and then evaluates the model's predictions on these perturbed samples. The importance of each word was quantified based on its contribution to the predicted class. This process was repeated for each input sample, whereas the results were aggregated to identify the most influential words across the dataset.

3. Experimental Work

The hybrid models developed in this study were deployed using Python. The performance of the models was evaluated using real-world datasets of fake news. The proposed models were deployed on the AraBERT pretrained model and three well-known deep learning algorithms, namely, CNN, LSTM, and GRU. However, to enhance their overall performance, a combination of three models namely, AraBERT with LSTM (AraBERT-LSTM), CNN with LSTM (CNN-LSTM), and CNN with GRU (CNN-GRU) were proposed. AraBERT is a groundbreaking transformer-based language model. It is specifically designed for Arabic natural language processing. This model, which was pre-trained on an Arabic-specific corpus, addresses linguistic issues with dialectal variations and semantic nuances, and its bidirectional encoding approach provides a better understanding of context. The hybrid model combines ArabBERT with LSTM as shown in Figure

3. The proposed model harnesses the advantages of the transformer and recurrent neural network architectures, where ArabBERT offers a comprehensive understanding of Arabic linguistic features, including intricate morphological and semantic details encoded in pre-trained embeddings, while the LSTM layer enhances the model with sequential memory capabilities to process temporal dependencies and extract long-range contextual information. This hybrid architecture is especially effective for processing Arabic text, which has a complex morphological structure, and enables greater language understanding, better generalization to other Arabic text domains, and improved performance by combining the global contextual awareness of transformers with the sequential processing strengths of recurrent neural networks.

3.1. The Implementation Environment

Google Colab provides the ability to conduct com Arrays of GPUs (Tesla T4, 16 GB memory) accelerated plex and computationally expensive experiments. The model is built using a Python library for deep learning (i.e., Keras) on top of the popular, widely used library, TensorFlow, with other python libraries were used to help data pre-processing, such as Pandas and NumPy. NLP tasks were performed using NLTK python library, whereas Scikit-learn is a powerful python library for ML used for data preparation, results assessment, and baseline classifications.

3.2. Model Implementation

The AraBERT-LSTM model uses a state-of-the-art deep learning architecture with hyperparameters fine-tuned to perform well for Arabic language processing tasks, using the Arabic Fake News Dataset (AFND) that incorporates AraBERT pre-trained transformer with a bidirectional LSTM layer (768-dimensional embeddings and 128 LSTM units of two layers), trained with Adam optimizer (learning rate 2e-5), batch size 8, weight decay 0.01, and regularization (dropout 0.3 in dense layers and L2 0.001). The model was tokenized with a maximum sequence length of 512 tokens, ReLU activation for hidden layers and SoftMax for the output layer. The model is trained for seven epochs based on observations of convergence and resource availability. Accuracy, precision, recall, and F1-score are computed to evaluate the model performance. The main steps of the proposed hybrid model are presented in Algorithm 1. The accuracy was calculated for all prediction classes. Equation 10 shows the accuracy formula for evaluating the total number of correct predictions. The recall (see Equation 11) is another metric for model's performance, which measures the right prediction class that corresponds to the label of the ground truth data. The precision (Equation 12) or positive prediction value (PPV) is an additional statistic measurement that evaluates the accuracy of positive predictions.

$$ACC = TP/n \quad (10)$$

$$Recall = TP/(TP+FN) \quad (11)$$

$$Precision = TP/(TP+FP) \quad (12)$$

where n is the total number of all classes and TP is the number of true or accurate prediction (True Positive) classes that the model successfully identified. The false positive rate (FP) is the number of false prediction classes that the model fails to match to the proper class in the ground truth data.

3.3. Algorithm 1: The BERT-BiLSTM Classification Model

1. Input: Arabic Text Dataset
2. Output: Predicted Labels (e.g., Fake/Real News)
3. Preprocessing:
   - Tokenizing text, using the BERT tokenizer to generate input IDs and attention masks.
   - Splitting the dataset into training and testing sets.
4. Model Architecture Initialization:
   - Loading the pre-trained BERT model for contextualized token embeddings.
   - Initializing a bidirectional LSTM layer to capture sequential dependencies.
   - Defining dropout and fully connected layers for classification.
5. Training Phase:
   - Forwarding Pass:
     - Feeding input IDs and attention masks into BERT to extract contextual embeddings.
     -Passing BERT embeddings through the bidirectional LSTM to model temporal features.
     - Applying dropout to the final LSTM hidden state for regularization.
     - Computing logits via the fully connected layer.
   - Optimization:

- Training the model end-to-end using cross-entropy loss and gradient descent.
- Updating weights for BERT, LSTM, and classification layers jointly.

6. Testing Phase:
   - Preprocessing test data using the BERT tokenizer.
   - Generating predictions by passing test data through the trained BERT-LSTM model.
7. Evaluation:
   - Measuring classification performance using accuracy, F1-score, precision, and recall.
   - Comparing results against baseline models to validate effectiveness.

This algorithm integrates BERT for contextual embeddings and bidirectional LSTM for sequence modelling, enabling robust feature extraction and classification in a unified architecture.

3.4. The Research Datasets

The Arabic Fake News Dataset (AFND) was used. It is a comprehensive corpus of Arabic digital textual content, widely used in research focuses on Arabic language misinformation [30]. The dataset was collected from 134 different sources, and it consists of 606,912 Arabic articles for robust computational linguistic analysis. The dataset was initially categorized into three categories: credible, incredible, and undecided. Since the dataset class is binary, the creators of the AFND dataset removed the undecided category, leaving 207310 and 167233 credible and incredible articles respectively, as shown in Figure 4.

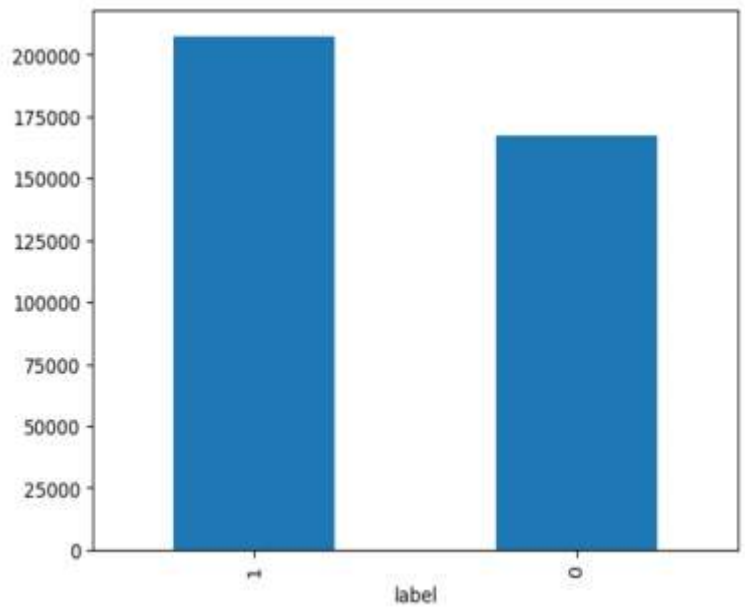

Figure 4: AFND dataset label volumes

The extracted text undergoes a series of systematic transformations, including the removal of punctuation and linguistic standardization techniques to standardize the textual representation. These preprocessing processes aim to reduce the risk of computational noise, increase the signal-to-noise ratio (SNR) in later machine learning processes, and preserve inherent semantic relationships between words, so that words can be represented as multi-dimensional real-valued vectors in a word-embedding layer, thus allowing for more sophisticated computational understanding of linguistic content. By implementing these methodological strategies, the research establishes a robust computational framework for Arabic language misinformation detection. This demonstrates robust approaches for dataset preparation and feature representation in the machine learning contexts. Accordingly, high performance could be achieved.

3.5. The Dataset split and pre-processing

The raw text was pre-processed to fit the requirements of text analysis. The first pre-processing task includes term steaming. This is to reduce each word to its root by excluding prefixes and suffixes, so this helps in processing similar words.

Another task is the remove Arabic stop-word, and unrelated contents like IP and URL address. The dataset then divided into two mutually exclusive sub-datasets, with 80% for the training phase, and 20% for the testing phase.

4. Results and discussion

Eight epochs later, the AraBERT-LSTM model's accuracy (ACC) was 98.16%. The validation model accuracy is displayed in Figure 5. There is little doubt that as the number of epochs rose, its accuracy increased dramatically. The model was tested on unseen data in order to learn more and comprehend its performance. These data were retained for evaluation reasons and were not used in the training phase. The confusion matrix provides a statistical summary of the model test outcomes (see Figure 6). It details the proportion of accurate and inaccurate forecasts for every class. The prediction class is displayed in the raw confusion matrix, and the true labels in the ground truth data are indicated in the columns. These outcomes show how well the suggested model performs with unknown data.

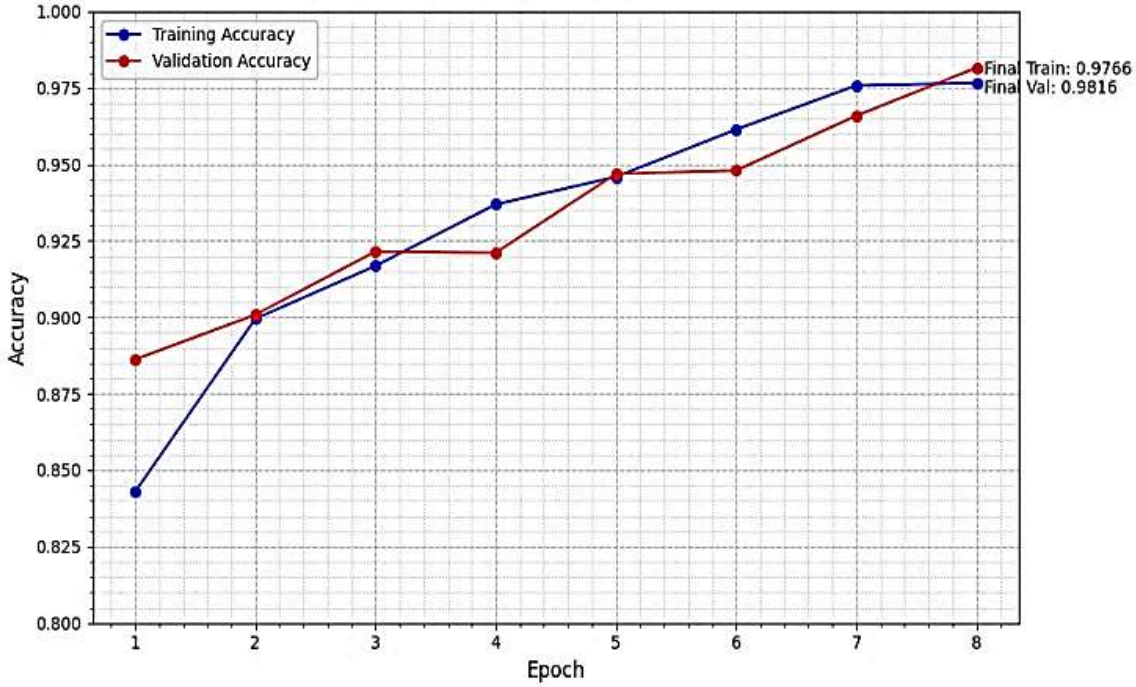

Figure 5: The training and validation model accuracy over epochs

The combination of transformer-based and BiLSTM architectures in AraBERT significantly enhances its ability to capture long-range dependencies and contextual relationships in Arabic fake news detection. This hybrid approach leverages the strengths of both models, allowing for improved understanding of complex linguistic patterns inherent in Arabic text. AraBERT utilizes transformer models to capture rich contextual information, which is crucial for understanding the nuances of Arabic language and culture. This study shows that transformer models outperform traditional deep learning models in various tasks, achieving high accuracy rates (up to 98%).

The integration of LSTM allows the model to retain long-term dependencies within the text, which is essential for understanding the flow of information in news articles. LSTM's capability to process sequences helps manage the complexity of Arabic morphology and this, in turn, can lead to improving overall classification accuracy. Figure 6 shows the confusion matrix, which summarizes the model's prediction for unknown data. The power of the proposed AraBERT-BiLSTM model was validated either in combination or separately. In Figure 5, the validation accuracy progression of various deep learning models across seven training epochs is presented.

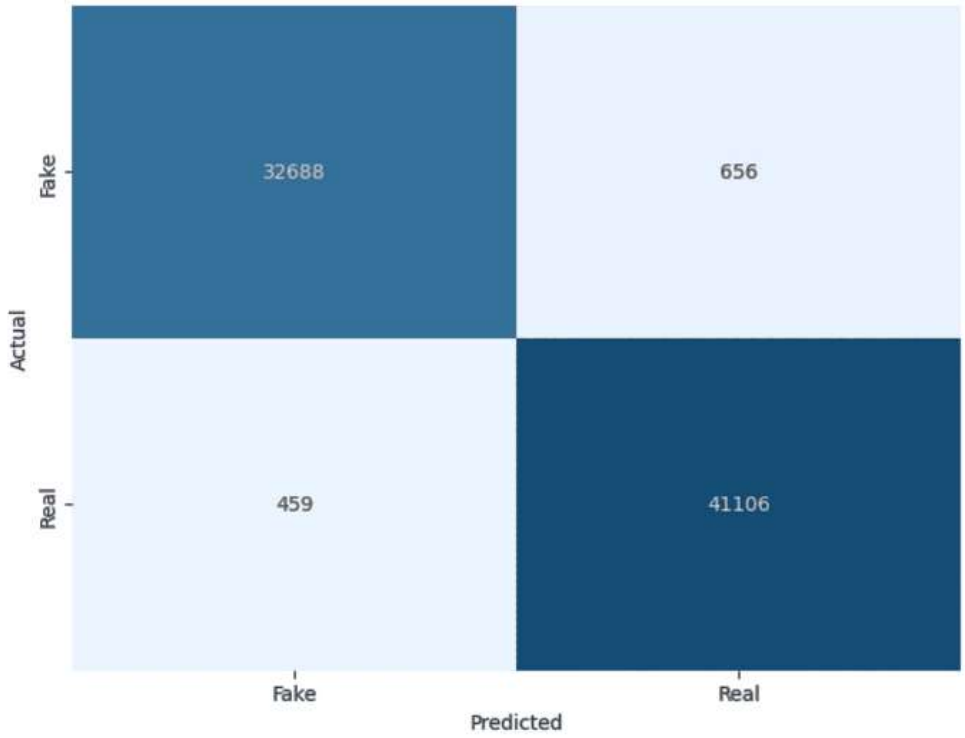

Figure 6: The confusion matrix

Table 2 illustrates the measures used to assess the proposed model.

Table 2: A comparison of the proposed models with original models.

Model	Accuracy	Precision	Recall	F1-score	Training Time
BiLSTM	88.63%	88.31%	87.88%	88.09%	1h 22m
BiGRU	89.7%	90.34%	87.80%	89.05%	1h 19m
CNN	72.87 %	73.99	73.87	72.48	17m
MARBERTv2	89.88%	0.9032	0.9146	0.9089	2h 10m
AraELECTRA	85.06%	0.8572	0.8751	0.8661	45m
AraBERT	88.19%	89.42	86.68	88.03	1h 30m
The proposed hybrid CNN-BiLSTM	91.73%	92.0%	92.0%	92.0%	1h 2m
The proposed hybrid CNN-BiGRU	92.0%	93.36%	90.74%	92.19%	58m
The proposed hybrid AraBERT-LSTM	98.16%	98.16%	98.16%	96.58%	7h 19m

The AraBERT-BiLSTM model shows a clear trend where the validation accuracy starts low (≈0.88) at the beginning of the training process and gradually improves to a level comparable to that of the other models (0.981) by epoch 7 (training time was approximately 7 hours and 19 minutes), indicating that combining the pre-trained transformer embeddings (AraBERT) with bidirectional LSTM layers for sequential learning is effective. On the other hand, CNN-BiLSTM and CNN-BiGRU hybrids show earlier convergence (0.91–0.92 by epoch 7), and converged within less than 1 hour, which may be attributed to the localized feature extraction of CNNs that improves the early training efficiency. However, when this is compared to AraBERT-BiLSTM, their prior performance revealed limitations in capturing long-range contextual dependencies in the absence of pre-trained embeddings. Strong baseline results were produced by the pre-trained standalone transformer models. MARBERTv2 was the model that converged with the highest accuracy (89.88%) and recall (0.9146), which took 2 hours and 10

minutes. AraBERT also did well with 88.19% accuracy and a high precision of 89.42%, training for approximately 1 hour and 30 minutes. AraELECTRA showed slightly lower metrics of 85.06% accuracy, training in only 45 minutes, but at a corresponding loss of performance. This shows that there is a clear trade-off between speed and predictive power among these architectures. Pure bidirectional architectures (BiLSTM, BiGRU) achieve moderate accuracy of 0.88 and 0.89, respectively, and take ~1h 22m and ~1h 19m, highlighting the benefits of hybrid designs. The findings highlight the balance between computational efficiency (faster convergence in CNN-based models) and contextual modelling (superior final performance in AraBERT-BiLSTM), influencing model choice depending on task-specific considerations such as resource constraints or high-precision needs. Table 3 compares the findings of the proposed model with other works on Arabic fake news detection. We use LIME to explain the predictions of the proposed hybrid model in this study, as LIME is an efficient method to explain the predictions of complex machine learning models such as deep neural networks by generating local explanations to identify the most influential features (words in our case) that contribute to the model decision-making process. LIME was used on 1,000 Arabic news articles, which were classified as genuine or fake by the hybrid AraBERT+BiLSTM model. A local explanation was generated for each article based on the words that most influenced the model prediction, and the top 20 influential words for the entire dataset were aggregated (see Table 4). The dataset has 1,000 samples, with 475 genuine and 525 fake news articles. The model had an average confidence score of 0.971, which indicates that the model is highly confident of its predictions.

Table 3: A comparison of the proposed model performance with previous literature

Author	Algorithm	Accuracy %
[31]	Bi-LSTM- BiGRU	89.5
[32]	LSTM-CNN	81
[23]	WaraBERT-V1	93.83
The proposed models	CNN-BiLSTM	91.73
	BiLSTM-BiGRU	92
	AraBERT-BiLSTM	**98.16**

Table 4 The most influential words based on LIME

Feature	Meaning	Count	Total importance	Average importance
ان	"that" or "if"	178	12.239	0.069
الي	"to me" or "my"	154	8.989	0.058
اليوم	today	112	10.470	0.093
كورونا	COVID-19	43	2.932	0.068
اخبار	News	32	3.614	0.113
وزير	minister	29	1.645	0.057
عبر	cross	28	1.752	0.063
مصر	Egypt	27	2.051	0.076
الصحة	health	27	1.726	0.064
وزارة	ministry	26	2.451	0.094
امس	yesterday	24	2.208	0.092
قطر	Qatar	24	1.568	0.065
الجزار	Butcher	23	2.710	0.118
مجلس	council	23	1.236	0.054
دنيا	world	21	6.155	0.293
مارس	March	19	1.244	0.065
المصدر	source	18	1.785	0.099
العربية	Arabic	18	0.648	0.036
الحكومة	government	17	1.576	0.093
السعودية	Saudi Arabia	17	1.463	0.086

4.1. Most Common Features

The word ان (that or if) appeared in 178 samples and had a total importance of 12.239 (12.239 + 0.069 × 178) and average importance per occurrence of 0.069, while الي (to me or my) appeared in 154 samples and had a total importance of 8.989 (8.989 + 0.058 × 154) and an average importance of 0.058. يوم (day) appeared in 112 samples with a total importance of 10.470 (10.470 + 0.093 × 112) and an average importance of 0.093.

4.2. Domain-Specific Words

Words such as كورونا (COVID-19), الصحة (health), and the proper noun الجزار (a proper noun that could be a person or place) were the most influential in certain contexts, such as (كورونا, Covied-19) in 43 samples with a total importance of 2.932 (reflecting its relevance in news related to the pandemic), and دنيا (world) had the highest average importance (0.293) among the top 50 words, despite appearing in only 21 samples (indicating that its presence significantly impacted the model's predictions in those samples).

4.3. Geopolitical and Institutional Terms

Words such as "مصر" (Egypt), "وزارة" (ministry), and "الحكومة" (government) were frequently influential, reflecting the model's sensitivity to geopolitical and institutional contexts. The word "السعودية" (Saudi Arabia) appeared in 17 samples with a total importance of 1.463, indicating its relevance in news related to the region.

4.4. Temporal Terms:

Temporal words, such as "**اليوم**" (today), "**امس**" (yesterday), and "**مارس**" (March) were significant, indicating that temporal context may be a factor in determining whether news is real or fake. Explaining predictions made by the hybrid AraBERT+BiLSTM model using LIME helps identify features that are important for the predictions, and reveals that both common stop words and domain-specific terms are influential, emphasizing the need for contextual information in fake news detection, and can be used to inform further enhancements to the model, such as fine-tuning preprocessing steps to handle stop words or incorporating domain-specific knowledge.

5. Conclusions

This study explored a variety of deep learning algorithms for detecting Arabic fake news, including CNN, BiGRU, BiLSTM, MARBERTv2, AraELECTRA, and AraBERT individually or in a hybrid combination. It compared the performance of independent and combined (hybrid) deep learning models. The results confirmed the scientific validity of integrating transformer-based contextual embeddings with sequential learning techniques to overcome the challenges of Arabic morphology and dialectal diversity. Therefore, the study provided a robust model that can handle dialectal differences and morphological diversity. The proposed model achieved the highest accuracy (98.16%) when compared to other algorithms, thus showing the significant improvements and robustness of the proposed model, while the individual BiLSTM algorithm had the lowest accuracy (88.63%). Thus, it was shown that combining deep learning methods with the AraBERT model is useful in identifying fake news in Arabic. Regardless of the significant findings of this research, it is not without limitations that may invite further research. The computational cost of the proposed model was high. Hence, further research is required by incorporating transformer-based components to improve the model's processing capabilities and reducing the computational cost. Moreover, the integration of attention mechanisms is required to help focus on relevant features. Finally, the improvement of dialectal Arabic processing can better handle the various Arabic language variants found in news content across different regions.

Conflicts of Interest: The authors declare no conflict of interest.

Funding Statement: The authors confirm that they have not received any funds.

References

[1] Kumar, S.; Singh, T.D.; "Fake News Detection on Hindi News Dataset". Glob. Transit. Proc., 3(1): 289–297, 2022. https://doi.org/10.1016/j.gltp.2022.03.014.

[2] Obayes, H.K.; Alhussayni, K.H.; Hussain, S.M.; "Predicting COVID-19 Vaccinators Based on Machine Learning and Sentiment Analysis". Bull. Electr. Eng. Inform., 12(3): 1648–1656, 2023. https://doi.org/10.11591/eei.v12i3.4278.

[3] Pierri, F.; Ceri, S.; "False News on Social Media: A Data-Driven Survey". SIGMOD Rec., 48(2): 18–27, 2019. https://doi.org/10.1145/3377330.3377334.

[4] Nassif, A.B.; Elnagar, A.; Elgendy, O.; Afadar, Y.; "Arabic Fake News Detection Based on Deep Contextualized Embedding Models". Neural Comput. Appl., 34(18): 16019–16032, 2022. https://doi.org/10.1007/s00521-022-07206-4.

[5] Alruily, M.; "Classification of Arabic Tweets: A Review". Electronics, 10(10): 1143, 2021.

https://doi.org/10.3390/electronics10101143.
[6] Albtoush, E.S.; Gan, K.H.; Alrababa, S.A.; "Fake News Detection: State-of-the-Art Review and Advances with Attention to Arabic Language Aspects". PeerJ Comput. Sci., 11: e2693, 2025. https://doi.org/10.7717/peerj-cs.2693.
[7] Abbas, M.A.; Abd, D.H.; Frikha, M.; Alimi, A.M.; "Arabic Fake News Detection Techniques: A Review". J. Intell. Syst. Internet Things, 18(1): 150–168, 2026.
[8] Hamadouche, K.; Bousmaha, K.Z.; Amar, M.Y.B.; Hadrich, B.L.; "Detection of Arabic and Algerian Fake News". Appl. Comput. Syst., 29(2): 14–21, 2024. https://doi.org/10.2478/ACSS-2024-0017.
[9] Alnabrisi, I.; Saad, M.; "Detect Arabic Fake News Through Deep Learning Models and Transformers". Expert Syst. Appl., 251: 123997, 2024. https://doi.org/10.1016/J.ESWA.2024.123997.
[10] 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(1): 5516945, 2021. https://doi.org/10.1155/2021/5516945.
[11] Shehata, A.; Al-Suqri, M.N.; Elshaiekh, N.E.; Hamad, F.; Alhusaini, Y.N.; Mahfouz, A.; "ArabFake: A Multitask Deep Learning Framework for Arabic Fake News Detection, Categorization, and Risk Prediction". IEEE Access, 12: 191345–191360, 2024. https://doi.org/10.1109/ACCESS.2024.3518204.
[12] Wotaifi, T.A.; Dhannoon, B.N.; "Improving Prediction of Arabic Fake News Using Fuzzy Logic and Modified Random Forest Model". Karbala Int. J. Mod. Sci., 8(3): 477–485, 2022. https://doi.org/10.33640/2405-609X.3241.
[13] Almandouh, M.E.; Alrahmawy, M.F.; Eisa, M.; Elhoseny, M.; Tolba, A.S.; "Ensemble Based High Performance Deep Learning Models for Fake News Detection". Sci. Rep., 14(1): 26591, 2024. https://doi.org/10.1038/S41598-024-76286-0.
[14] Othman, N.A.; Elzanfaly, D.S.; Elhawary, M.M.M.; "Arabic Fake News Detection Using Deep Learning". IEEE Access, 12: 122363–122376, 2024. https://doi.org/10.1109/ACCESS.2024.3451128.
[15] Matti R. S.; Yousif S. A.; "Leveraging Arabic BERT for High-Accuracy Fake News Detection". Iraqi J. Sci., 66(2): 751–764, 2025. https://doi.org/10.24996/ijs.2025.66.2.18
[16] Azzeh M.; Qusef, A.; Alabboushi, O.; "Arabic Fake News Detection in Social Media Context Using Word Embeddings and Pre-trained Transformers ". Arab. J. Sci. Eng. 50(2), 923–936, 2025. https://doi.org/10.1007/s13369-024-08959-x.
[17] El-Mageed, S.M.; Aboutabl, A.E.; Mohamed, E.H.; "ArFakeDetect: A Deep Learning Approach for Detecting Fabricated Arabic Tweets on COVID-19 Vaccines". In: 2024 Intelligent Methods, Systems, and Applications (IMSA), Giza, Egypt, 13-14 July; Gomaa, W.; Ghoneimy, M.; El-Sersey, W.; IEEE: Piscataway, NJ, USA, 2024. https://doi.org/10.1109/IMSA61967.2024.10652806.
[18] Aboulola, O.I.; Umer, M.; "Novel Approach for Arabic Fake News Classification Using Embedding from Large Language Features with CNN-LSTM Ensemble Model and Explainable AI". Sci. Rep., 14: 30463, 2024. https://doi.org/10.1038/s41598-024-82111-5.
[19] Fares, A.; Chougui, R.Y.; Drif, A.; Giordano, S.; "An Ensemble Deep Learning Models Based on Metadata for Measuring Arabic Fake News Uncertainty". In: 2024 IEEE International Conference on Advanced Systems and Emergent Technologies (IC_ASET), Hammamet, Tunisia, 27-29 April; Ben Amor, A.; Bhar Elayeb, S.; Ghorbel, C.; Elloumi, S.; Nouri, K.; Naceur, M.S.; IEEE: Piscataway, NJ, USA, 2024.
[20] Rane, N.L.; Choudhary, S.P.; Rane, J.; " Ensemble Deep Learning and Machine Learning: Applications, Opportunities, Challenges, and Future Directions". Stud. Med. Heal. Sci, 5(2): 18–41, 2024. https://doi.org/10.48185/smhs.v1i2.1225.
[21] Antoun, W.; Baly, F.; Hajj, H.; "AraELECTRA: Pre-Training Text Discriminators for Arabic Language Understanding". In: Proceedings of the Sixth Arabic Natural Language Processing Workshop, Kyiv, Ukraine (Virtual), 21 April; Habash, N.; Bouamor, H.; Hajj, H.; Magdy, W.; Zaghouani, W.; Bougares, F.; Tomeh, N.; Ibrahim A.; Samia T.; Association for Computational Linguistics: Stroudsburg, PA, USA, 191–195, 2021.
[22] Turki, H.M.; Al Daoud, E.; Samara, G.; Alazaidah, R.; Qasem, M.H.; Aljaidi, M.; Abuowaida, S.; et al.; "Arabic Fake News Detection Using Hybrid Contextual Features". Int. J. Electr. Comput. Eng., 15(1): 836–845, 2025. https://doi.org/10.11591/ijece.v15i1.pp836-845.
[23] Kwaik, K.A.; Chatzikyriakidis, S.; Dobnik, S.; Saad, M.; Johansson, R.; "An Arabic Tweets Sentiment Analysis Dataset (ATSAD) Using

Distant Supervision and Self Training". In: Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, Marseille, France, May; Al-Khalifa, H.; Magdy, W.; Darwish, K.; Elsayed, T.; Mubarak, H.; Eds.; European Language Resource Association: Paris, France, 1–8, 2020.

[24] Antoun, W.; Baly, F.; Hajj, H.; "AraBERT: Transformer-Based Model for Arabic Language Understanding". In: Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, Marseille, France, May; Al-Khalifa, H.; Magdy, W.; Darwish, K.; Elsayed, T.; Mubarak, H.; Eds.; European Language Resource Association: Paris, France, 9–15, 2020.

[25] Wu, T.W.; Zhang, H.; Peng, W.; Lü, F.; He, P.J.; "Applications of Convolutional Neural Networks for Intelligent Waste Identification and Recycling: A Review". Resour. Conserv. Recycl., 190: 106813, 2023. https://doi.org/10.1016/j.resconrec.2022.106813.

[26] Hochreiter, S.; Schmidhuber, J.; "Long Short-Term Memory". Neural Comput., 9(8): 1735–1780, 1997.

[27] Waqas, M.; Humphries, U.W.; "A Critical Review of RNN and LSTM Variants in Hydrological Time Series Predictions". MethodsX, 13: 102946, 2024. https://doi.org/10.1016/j.mex.2024.102946.

[28] Khalil, A.; Jarrah, M.; Aldwairi, M.; Jaradat, M.; "AFND: Arabic Fake News Dataset for the Detection and Classification of Articles Credibility". Data Brief, 42: 108141, 2022. https://doi.org/10.1016/j.dib.2022.108141.

[29] Syed, L.; Alsaeedi, A.; Alhuri, L.A.; Aljohani, H.R.; "Hybrid Weakly Supervised Learning with Deep Learning Technique for Detection of Fake News from Cyber Propaganda". Array, 19: 100309, 2023. https://doi.org/10.1016/j.array.2023.100309.

[30] Sorour, S.E.; Abdelkader, H.E.; "AFND: Arabic Fake News Detection with an Ensemble Deep CNN-LSTM Model". J. Theor. Appl. Inf. Technol., 100(14): 5072–5086, 2022.