

Deep Learning Techniques To Detect Arabic Fake News

Enas Tariq Khudair^{1*}, Onsa Lazzez¹, Mourad Zaied², Tarek M. Hamdani³, Ahmed T. Sadiq⁴,
Habib Chabchoub⁵, Adel M. Alimi^{1,6}

¹REGIM-Lab.: REsearch Groups in Intelligent Machines, University of Sfax, University of Sfax,
National Engineering School of Sfax (ENETCOM), Sfax, 3038, Tunisia.

²RTM-LAB, University of Gabes, Gabes, Tunisia.

³Higher Institute of Computer Science Mahdia (ISIMa), University of Monastir, Tunisia, Monastir
City, 10587, Tunisia Country.

⁴Department of Computer Science, College of Science, University of Technology-Iraq, Baghdad,
Iraq.

⁵College of Business, Al Ain University, Abu Dhabi, United Arab Emirates.

⁶Department of Electrical and Electronic Engineering Science, Faculty of Engineering and the
Built Environment, University of Johannesburg, Johannesburg 2006, South Africa.

*Corresponding author(s). Email(s): Enas.T.Khudir@uotechnology.edu.iq;

Contributing authors: Onsa.lazzez@enis.tn; mourad.zaied@univgb.tn; tarek.hamdani@enis.tn;
Ahmed.t.sadiq@uotechnology.edu.iq; hbib.chabchoub@aaau.ac.ae; adel.alimi@enis.tn;

These authors contributed equally to this work.

Abstract

Deep learning (DL)-based detection systems for fake news should improve and adapt as fake news grows more complex in order to continue protecting the integrity of information in digital society. In addition to addressing a serious technological challenge, creating strong DL-based fake news detection systems is an important tool to maintain authenticity of information. Because of the difficulty of finding Arabic data, the method offers a basis for understanding the significance and difficulties of collecting data in Arabic. Due to the limited dataset, the suggested system has been translated from English into Arabic, and its performance and the potential of DL in identifying fake news were verified using another dataset that was accessible. Existing studies utilizing DL methods, like recurrent neural networks (RNNs), LSTMs, and convolutional neural networks (CNNs) for detecting fake news are discussed. For comparison, the term Aribert is used. It makes use of cutting-edge DL methods to propose a new method for identifying fake news. The paper offers a thorough framework for automatically detecting as well as classifying any false information on digital platforms by combining deep neural networks (DNNs) with natural language processing (NLP). Contextual understanding, temporal dependency, and content complexity are some of the main challenges with fake news detection that the approach solves. From purposefully created news articles to thinly veiled misleading content, the system displays great performance in detecting various types of misinformation. Furthermore, cutting-edge feature extraction (FE) methods have been used, which take into account metadata as well as textual content, such as propagation patterns and source credibility. When put to comparison with machine learning (ML) methods, the experimental results show notable gains in detection speed and accuracy. Spacey, Fasttext, and two-word embeddings were among the four methods that were employed. The best DNN method used Spacey to obtain a strong performance of 78%, whereas LSTM model performed well with Spacey at 49% and Fasttext at 51%. On accuracy scale, it scored a 99.1% success rate after converting to BERT, whereas AraBERT earned a 99.3% success rate. AraBERT is thus better. This is due to the fact that both AraBERT and BERT are trained on various Arabic articles.

Keywords: Deep learning, Fake news detection, CNN, DNN, RNN, Hybrid LSTM and AraBERT, Misinformation, and Social Media Analysis.

1. Introduction

One of the significant and challenging problems is fake news, which faces numerous obstacles. Fake news is presently being studied because of the nature of languages and the challenge of spotting it. DL is a ML method that aims to depict the world as nested hierarchy of concepts that are identified by DL architecture automatically [1]. Specifically,

DL—also referred to as deep structured learning—is a subset of ML technology that learns representations through artificial neural networks (ANNs). Semi-supervised, supervised, and unsupervised learning are the three learning categories. DL approaches that use DNNs have gained popularity as a result of the development of high-performance computer facilities. In particular, DL has more flexibility and functionality because it could manage a large number of functions while working with unstructured data. In essence, DL feature algorithms transfer data through a number of layers, each of which has the ability to progressively extract features before passing them on to the next layer. Low-level features are extracted in the first layer, and features are combined in the next layers in order to provide a comprehensive representation. DL was developed inseparably with time, achieving a blast of information from all over the world and in all kinds of structures. This information, which is known as "large information," comes from various sources, which include online searches, social media, web-based commercial platforms, and online movies. Through fintech applications, such as distributed computing, this enormous amount of information is instantly accessible and could be shared.

DL can be defined as an artificial intelligence (AI) function mimicking the way that the human brain can process data and generate patterns for use in decision-making. In AI, it is a subset of ML that includes networks which can learn in unsupervised manner from unstructured data. To be more specific, DL encompasses a variety of networks, including CNN (Convolutional Neural Network), DBN (Deep Belief Network), Recursive Neural Network, RNN (Recurrent Neural Network), and a lot more. In order to achieve this, neural networks (NNs) are extremely helpful in sentence modelling, sentence classification, vector representation, word representation estimation, and future presentation [3].

A popular area of ML is DL, which draws inspiration from how the biological brain of humans handles information processing and draws conclusions. In order to improve accuracy for complex tasks, DL employs multiple layers of neurons, each of which is coupled to success layer. The use of an expert is reduced because to the development of DL from ML [4].

To make predictions, learn from data, and maximize performance, DL models are made up of a number of basic building components. Each of these elements—optimization algorithms, layers, activation functions, and loss functions—is essential to the model's capacity for learning from and generalizing from data [5].

With quickly developing applications spanning atomistic, image-based, textual, and spectral data modalities, DL is one of the materials data science subjects with the quickest rate of growth. It enables automated feature recognition and analysis of unstructured data. In this sense, the use of DL methods in atomistic prediction, has been stimulated by the recent growth of huge materials databases [6].

For various ML problems during the past 20 years, DL has provided the solution. Dependency on a large amount of labeled data and training costs are its two main drawbacks. Through reusing knowledge from a source data or task in training on a target data or work, transfer learning (TL) in DL, sometimes referred to as Deep Transfer Learning (DTL), aims to lower such cost and dependency. Most DTL approaches that are used are based upon networks or models [7].

Our contributions can be summarized below:

1. Reading and preparing the dataset and breaking the task down into manageable parts.
2. The first section used Spacy and Fasttext methods to perform preprocessing on the dataset.
3. The second section used transformer models such as BERT and AraBERT.

4. The third section used different deep learning methods, including RNN, CNN, and LSTM.
5. CNN and RNN were combined to get the results and compare them with the methods without merging.
6. The accuracy was adopted for evaluating the performance regarding the proposed system, combining the two methods, the Spacy and the Fasttext.

2. Related work:

Through utilizing ML-based fake news detection model, the authors in [8] offered a solution. Prerequisite data for the model must be taken from a variety of news websites. Data extraction, which is then utilized for creating datasets, was accomplished using the web scraping approach. True datasets and false datasets are the two main groups into which the data was categorized. Logistic Regression, Random Forest, KNN, Decision Tree, and Gradient Booster are the classifiers utilized to classify the data. Data was classified as either false or true depending on the result that has been obtained. The user could therefore determine on the webserver if the provided news is fake or not.

To detect the fake text, authors of [9] suggested many classifier types. In order to determine which classifier would be most effective for identifying fake news in Arabic, they next tested each of such classifiers using two distinct Arabic datasets. Fast Text library was used with all of such classifiers to improve text classification by offering effective word representations.

The authors of [10] employed a DL method that incorporated pre-trained word embedding that was trained with the use of four distinct datasets with many architectures, like the Bidirectional LSTM, CNN, and ResNet. For lessening data imbalances between the classes, every data undergoes a data augmentation procedure that utilizes back-translation method. The outcomes demonstrated that, across all evaluated datasets, the Bidirectional LSTM architecture performed better than ResNet and CNN.

In [11], the researchers used many cutting-edge AI methods to assess the created dataset. A total of five ML methods has been specifically employed: KNN, Naïve Bayes, SVM, Logistic Regression, and Decision Trees. GloVe and BERT embeddings, on the other hand, have been employed with two DL methods, LSTM and CNN. F1-score, precision, recall, and accuracy were used for comparing the performance regarding all applied models as well as embeddings. According to the results, the best performance was achieved with LSTM initialized with GloVe embeddings. Additionally, the study examined the misclassified samples through contrasting them with human judgements.

A thorough analysis of ML and DL-based methods for the detection of fake news was published in [12]. For practitioners and researchers interested in creating efficient fake news detection systems with DL and ML methods, their review offered insights and recommendations. Prior to reporting or publishing on a story, news reporters frequently need to confirm its authenticity. Reporters may eliminate fake news and concentrate on covering reliable and accurate information through using fake news detection models. The authors of [13] built DL models to identify fake news depending on news titles or content and used NLP methods for text analytics. The research's suggested solution seeks to be implemented in real-worlds social media platforms and get rid of the negative experience that users have when they acquire misleading stories from unreliable sources. Before vectorizing the text into N-gram vectors or sequence vectors with the use of terms frequency inverse document frequency (TF-IDF) or one-hot encoding, respectively, NLP methods employed text preprocessing methods, like regular expression, lemmatization, tokenization, and stop words removal. Tensor Flow was therefore selected as the framework to be utilized thanks to its integrated Keras DL libraries, which have a sizable

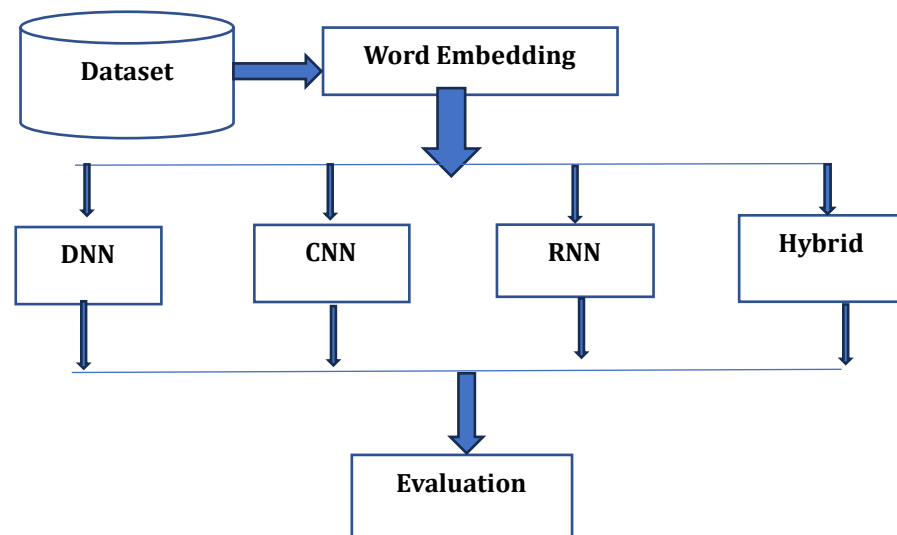
community and enough comments on its GitHub repository to support the construction of DL-NN models.

Since the main objective of fake news site is to sway public opinion on particular topics, the authors of [14] created an accurate and reliable model that can determine whether a news piece is fake or real.

3. The proposed Work

The primary function of the proposed system is to detect news written in Arabic, whether it is fake or not (positive or negative). The practical components of the system are explained in this section. The system consists of two main components: collecting a dataset and using Arabic text to analyze the news. In this regard, collecting the dataset is a challenging step. The second component of the system analyzes the Arabic text. Fake news detection aims to identify forgery conveyed in a given text. Here, the system prioritizes texts written in Arabic. To do this task, this section recommends using different methods, such as DL algorithms. The proposed system achieved the goal of AFN detection by employing deep learning methods. As shown in Figure (1), due to the limited availability of Arabic databases, the dataset was translated from English to Arabic and used with other datasets for validating the performance regarding the suggested system. The procedures included reading and preparing the dataset and dividing the task into manageable parts. The first part of the work used four approaches, including the Spacy and FastText methods. The second part of the work used transformed models such as BERT and AraBERT. The third part used various deep learning methods, including DNNs, CNNs, and LSTMs, as well as hybrid CNN-LSTM models. All of these models were created using Fasttext and Spacey embedding methods. The best approach for deep neural networks was the achieved DNN.

Figure 1: Block diagram of the suggested system



In the next subsections, each part of the proposed system architecture will explained:

3.1. Arabic Fake News Dataset

To develop our classification model, a Kaggle dataset has been used for fake news detection. English was the initial language used to write the dataset. But since the work

was conducted on an Arabic fake news detection feature, Python code and the "google-trans" package have been used for translating it into Arabic before exporting it as an Excel file. Our English-to-Arabic translator code was unable to translate several Russian and German characters in the dataset. As a result, such characters were eliminated as the initial preprocessing step, followed by the elimination of any English words that were still difficult to translate into Arabic, including terms and names (e.g., Donald Trump, Jennifer, etc.). Actually, 20,000 instances were first trained using Kaggle dataset. Following preprocessing, however, the dataset included 10,000 instances, with 20% going to testing and 80% to training, with 5,000 of those being fake and 5,000 being real. Arabic and English numbers have been also eliminated, in addition to superfluous punctuation and pauses. Lastly, a pre-trained model segmentation tool has been used in order to divide each sentence into tokens throughout the training phase.

4. Word Embedding:

One of the most notable developments in NLP is Word Embedding. By converting words into vector-based numerical representations, such methods seek to improve computer models' comprehension of word relationships and meanings. Word Embedding approaches like Spacy and Fasttext map each word to a point in a high-dimensional space, in which text with comparable meanings are closely related, rather than encoding words as text strings. This shift from textual to numerical representations improves efficiency on a number of tasks, which include text classification, sentiment analysis, and machine translation. By improving models' capacity to comprehend language and interpret meaning, Word Embedding approaches expand the potential applications of AI.

4.1. Fast Text Embedding

The Facebook AI Research team created the word embedding technology. Through adding subword information, it is intended to enhance conventional word representation models and produce better embedding, particularly for morphologically rich languages. Fast Text can comprehend and generate embeddings for morphologically complex languages by converting words down into a set of n-grams, which aids in understanding the internal structure regarding words. Conventional word embedding models, like Word 2Vec, rate a fixed vocabulary throughout training, which might cause problems with OOV words. In order to address this issue, Fast Text combines the embeddings of the n-grams that make up OOV words to create an embedding for them. This feature is especially helpful in practical applications where new words—like new slang and technical terms—occur often.

4.2. Spacy Embedding

It is a robust and advanced Python NLP library that combines high performance and ease of use, especially for production use. It is appropriate for a variety of NLP and understanding applications due to its extensive feature set, support for contemporary word embeddings, and deep feature learning models. Data scientists and developers that work with language data frequently choose it because it provides a broad range of functions for effective text processing as well as analysis. Tokenization, part-of-speech tagging, named entity recognition (NER), multilingual support, pipeline engineering, dependency analysis, named entity recognition (NER), word vectors and embedding, semantic analysis, text classification, and visualization are some of the key features.

5. The AraBERT Model

A pre-trained language model created especially for processing Arabic text, AraBERT can efficiently handle a variety of NLP tasks. With regard to language understanding, its BERT-

based architecture makes use of the most recent developments in DL. AraBERT offers a strong framework for creating complex language applications to scholars and practitioners engaged in Arabic NLP. Key elements of AraBERT include its pre-training on a large dataset of Arabic text and its comprehension of the language's semantics, syntax, and common phrases. It is intended to capture the distinctive complexities and characteristics regarding Arabic language, including its dialectal variances, morphology, and script. Similar to original BERT, AraBERT understands a word in the context of all other words in a sentence, making it especially effective for tasks needing accurate comprehension. In comparison to models trained from scratch, AraBERT could be optimized for particular follow-up tasks, such as named entity recognition, question answering, and sentiment analysis, greatly enhancing performance. There are various AraBERT versions, including AraBERTv2 and AraBERTv1, each of which was trained on a different dataset or optimized for a particular set of tasks. Users are able to select the version that best meets their requirements. Token classification, classification tasks, and sequence classification are just a few of the NLP applications that AraBERT could be used for. Developers and researchers can easily utilize AraBERT for both non-commercial and commercial uses because it is usually available in an open-source format.

6. Neural Networks:

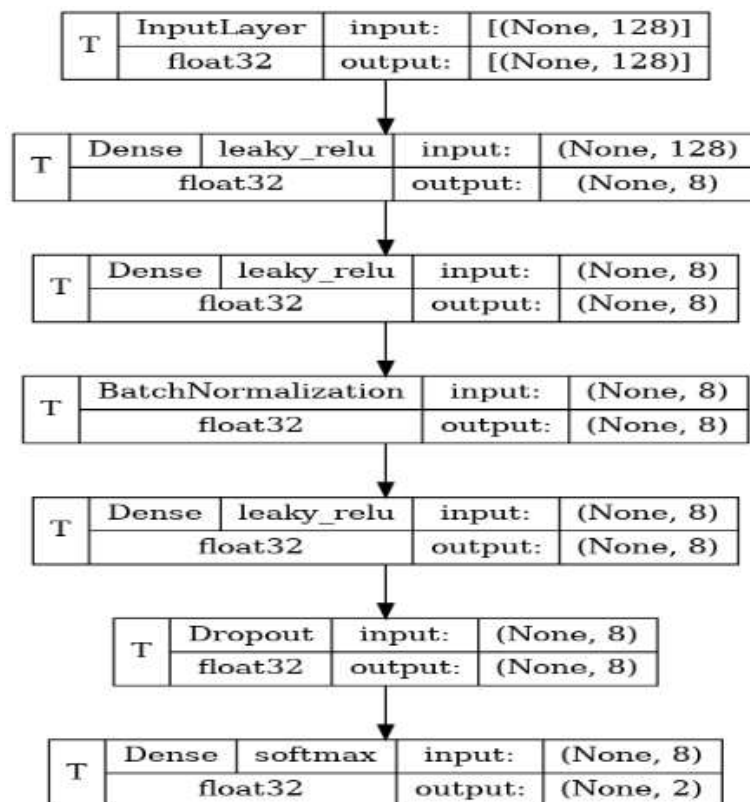
6.1. Deep Neural Networks (DNNs):

Advances in a variety of domains are made possible by DNNs, a versatile and powerful method for AI and ML. Through their capacity to represent complex relations among data, DNNs continue to propel technological and scientific advancements. Multiple hidden layers, an input layer, and an output layer make up DNNs. The ability for learning the complex features and patterns in data is made possible by the existence of several hidden layers. Neurons make up each layer of a DNN. After applying a weighted sum and adding a bias, each neuron processes the input from the preceding layer before passing it via a nonlinear activation function. The network can learn complex mappings between outputs and inputs by being fed such nonlinear activation functions. Data is fed into the input layer during the process of forward propagation, and it moves through each layer until reaching the output layer, in which classifications or predictions are made. DNNs learn by calculating the error between the actual and predicted outputs, a process known as backpropagation. In order to reduce error, the model after that modifies its weights in accordance with the loss gradient with respect to such weights. The degree to which the DNN's predictions and actual values agree is determined by the loss function. The cross-entropy loss for tasks of classification as well as the mean squared error regarding regression tasks are examples of common loss functions. To help the model converge to a minimum loss, the weights are updated during training using a variety of optimization algorithms. Since DNNs frequently identify hierarchical representations of data, they might automatically learn features from raw data without requiring a great deal of manual feature engineering. They could be used with unstructured data, time series, and structured data, among other data types and domains. DNNs are excellent for complex tasks because they could generalize well with enough data.

Figure (2) shows the structure of a DNN-based model. It consists of three layers: hidden, output, and input. Keras was used to define a neural network model with a serial API. Utilizing the leaky ReLU activation function, the initial dense layer consists of eight units. The training data's shape serves as the basis for the input shape. The leaky rule comprises eight units in the second thick layer. The input's shape is deduced from the layer before it,

the batch normalization layer, which aids in normalizing the inputs. Eight units comprise another dense layer and a Leaky Relu—leakage layer with leakage rate 0.5. Dropout is used as a regularization method. The output layer has two units and uses softmax activation. Particularly, this is a binary classification problem. The softmax function predicts the class that has the highest probability. Consequently, as neural network for binary classification with regularization methods has been created.

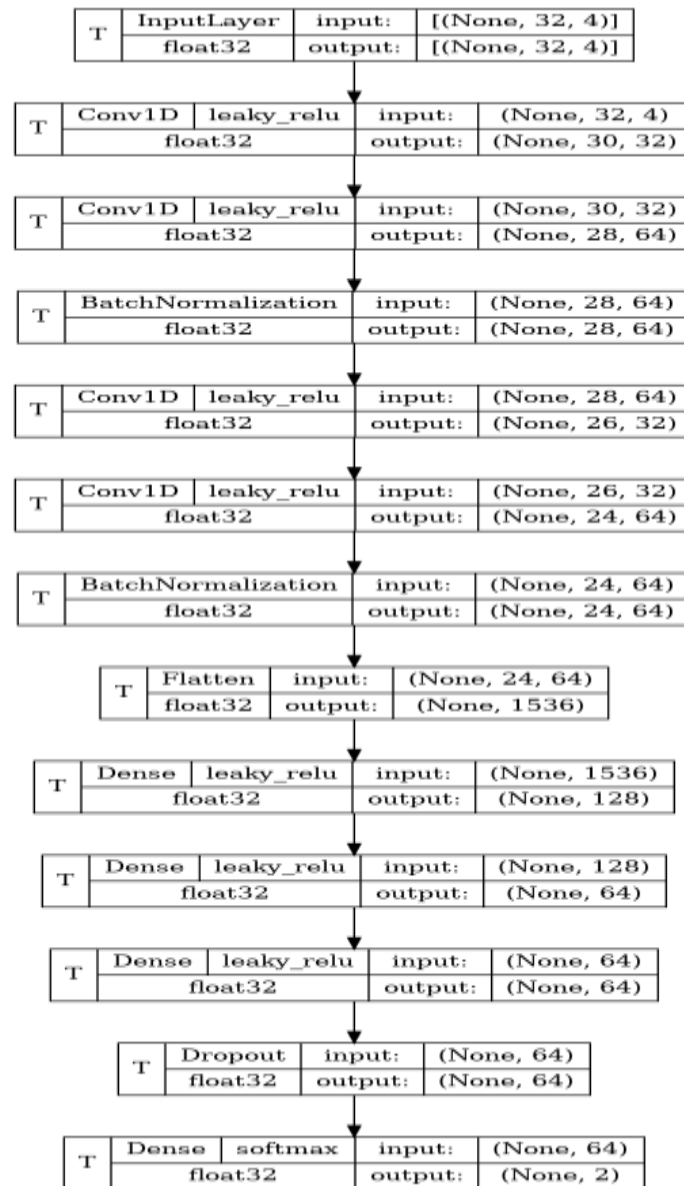
Figure 2: DNN Architecture



6.2. Convolution Neural Networks (CNNs):

CNNs are crucial to modern AI applications and have completely transformed the field of computer vision by offering strong instruments for effectively and efficiently evaluating visual data. They are a particular type of DL model made particularly for handling structured network data, especially images. They do exceptionally well in tasks like object detection, segmentation, and image classification. In contrast to typical NNs, which use convolution in place of global matrix multiplication, pooling layers, activation functions, and fully connected layers, such networks have locally applied convolutional layers. Due to their relatively basic network architecture, CNNs are crucial for DL since they require fewer parameters, which increases processing speed efficiency. CNNs could also help with speech and handwriting recognition, image analysis, and natural language understanding because they require little preprocessing. In the presented study, a custom CNN that is designed for particular sequence tasks is programmed using Keras and Sequence API.

Figure 3: CNN Architecture



6.3. Long Short-term Memory (LSTM):

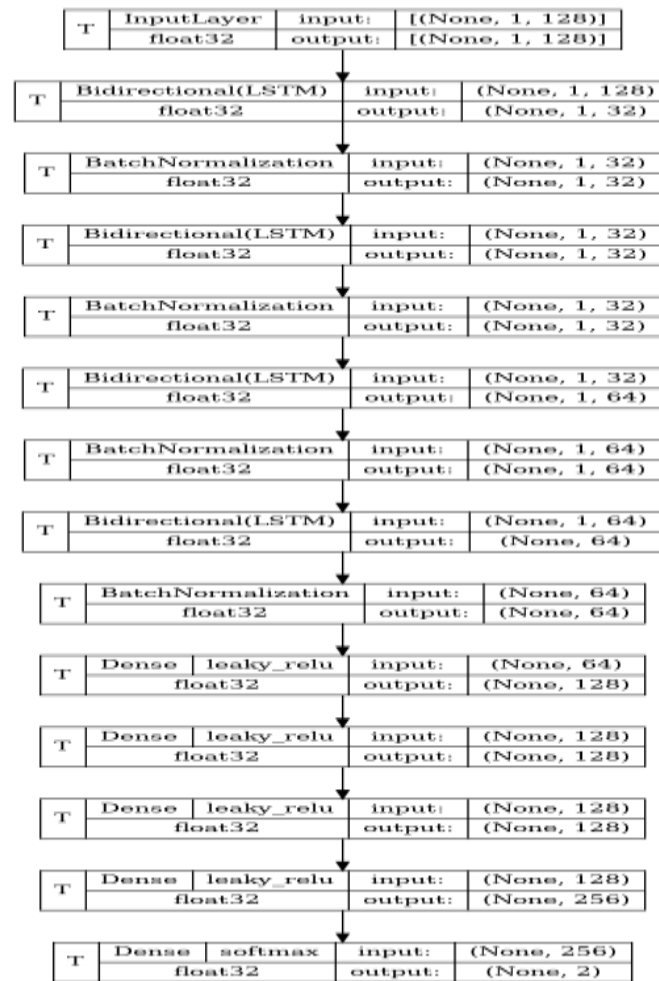
LSTM modules address the shortcomings of conventional RNNs and are an effective tool for sequential data analysis. They perform exceptionally well in various applications in speech recognition, time series prediction, and NLP due to their capacity to selectively forget and remember information. Long-term information retention is possible thanks to the memory cells found in LSTMs. More specifically, forget gates, input gates, and output gates are the three types of gates that LSTMs use to control information flow.

A more complex NN with bidirectional LSTM layers has been constructed using Keras and a serial API. Each layer of the model will be looked at separately: Sequences are returned by the first layer, a 16-unit bidirectional LSTM layer with leaky ReLU activation. The input shape for such layer is determined by the training data dimensions. A batch normalization layer comes after the first layer of LSM. The model also includes an additional 16-unit bidirectional LSTM layer that generates sequences. One more batch normalizing layer comes after this phase. The model's third LSTM layer consists of 32 bidirectional units that return sequences. A batch normalization layer comes after this layer. The fourth layer

of the LSTM simply provides the final state; sequences are not returned, despite having 32 units and being bidirectional.

This layer is followed by a batch normalization layer. Leaky ReLU activation and a dense layer with 128 units are included in the model. The model contains an additional thick layer with the same properties. Leaky ReLU is activated at 128 units by the third dense layer. A dense layer with 256 units and Leaky ReLU activation is also present. Specifically, the output layer of the model is suitable for binary classification due to its softmax activation and two units.

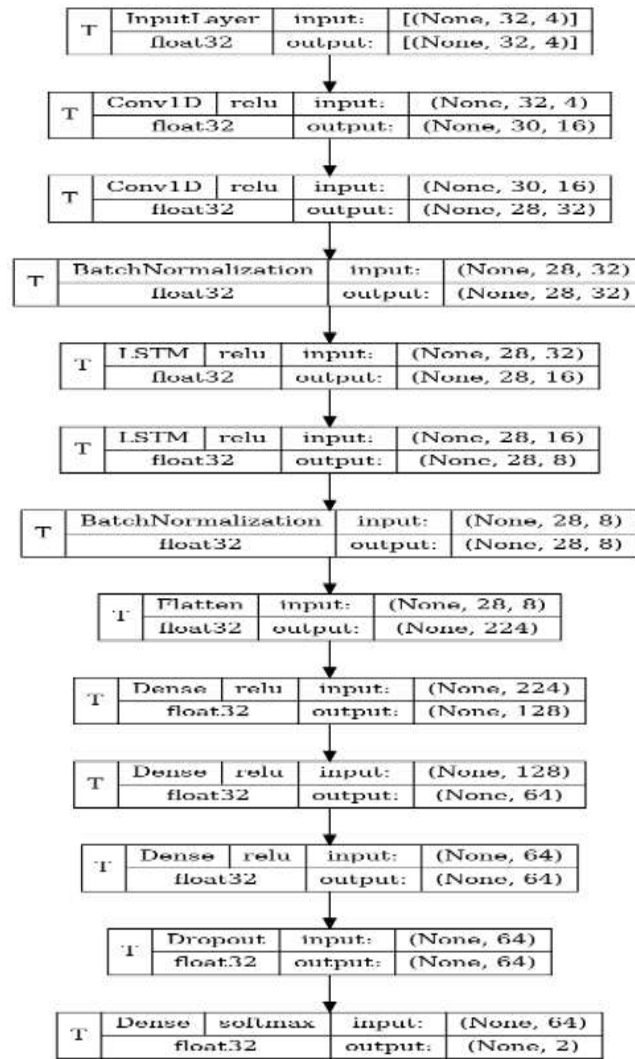
Figure 4: LSTM Architecture



6.4. Hybrid CNN and LSTM:

CNN-LSTM hybrid model effectively processes data with temporal and spatial dimensions by combining the advantages of LSTM networks with CNNs. Tasks where temporal features (like sequences of frames) and spatial features (like frames in a video) are crucial, such as event recognition, video analysis, and time series prediction, benefit greatly from this architecture. The model can better learn complex patterns that span both dimensions because of such combination.

Figure 5: CNN and LSTM Architecture



7. Experimental Results

This research faced the difficulties of collecting data in Arabic and choosing the best ones for this work. After that, different DL models have been used, with certain changes made, such as changing the number of layers and selecting the right design. Through testing, assessment, and comparison of several models, this procedure sought to show how to rate the most successful results. The models' word embeddings varied, and they were tested using a variety of approaches in an effort to outperform the findings of earlier studies. The focus has been directed towards optimization and quick identification of the most efficient and effective methods to achieve outstanding results. The best method was DNN using word embedding for spacy, achieving 78%, while DNN also achieved the highest performance for Fast text, achieving 76%. As for BERT, it achieved a 99.1% success rate, and AraBERT achieved 99.3% according to the accuracy scale. As a result, the AraBERT is the best method from our proposed deep learning, but after using transformer.

Table1:

Model	Accuracy%	
	Spacy	Fast Text
DNN	78%	76%
CNN	47%	46%

LSTM	51%	49%
Hybrid CNN+LSTM	54%	53%
BERT	99.1%	
AraBERT	99.3%	

As it has been mentioned earlier, the process of identifying fake news is not easy. To handle this issue, four DL models have been proposed to detect or identify fake news depending on the collected database. After completing the training process using two types of embedding, namely Spacy and Fast Text, the best method was the deep neural network. However, after resorting to more comprehensive methods such as BERT and AraBERT, it has been found that the best method is AraBERT, which achieved results of 99.3% based on the database on which it was trained. The reason is that this network was trained on a huge amount of data in the Arabic language, which gave it the strength in discrimination. It is currently considered one of the best bidirectional deep learning methods for the Arabic language.

8. Metrics

Several metrics have been calculated for the best approach (AraBERT) with a confusion matrix for this approach, as shown in the table below.

Table 2: Metrics for the AraBERT Approach

Accuracy	Precision	Racall	F1-Score
99.3%	99.44%	99.16%	49.64%

Table 3: Confusion Matrix for the AraBERT Approach

	Real	Fake
Real	4972	28
Fake	42	4958

These metrics selected the AraBERT, as it is the best method, and it is suitable for the Arabic language. Specifically, it was trained on data in the Arabic language.

9. Conclusion

The detection of fake news has drawn more attention recently because of its detrimental impacts and quick propagation. Given the limitation of existing literature, the detection of fake news in Arabic is a potential yet difficult study topic. Because there are no consistent features to differentiate between real and fake news, there was a trouble with the dataset and detecting fake news. The variety of orthography, dialects, rich vocabulary, and lack of Arabic databases are some of the obstacles that fake news detection in Arabic encounters. In order to determine if Arabic news is real or fake, a method for identifying fake news in Arabic was presented in this research. The English dataset was translated into Arabic, though, because there aren't many databases in this subject. The performance regarding the suggested system was then verified using another Arabic dataset that was accessible. Fasttext and Spacy were two of the four methods that have been employed with two words embedding. DNN, which used spacy to achieve 78% performance, was the best method. With a performance of 51% for Spacy and 49% for Fasttext, LSTM model achieved a good performance. On the accuracy scale, however, it scored a 99.1% success rate following converting to BERT, whereas AraBERT earned a 99.3% success rate. AraBERT is thus better. This is due to the fact that both AraBERT and BERT have extensive Arabic article training.

We propose that word embedding approaches with semantic support can be used in future work to support DL.

Author contributions

Author Contribution: Enas Tariq Khudair: Conceptualization, Methodology, and Writing-original draft. Onsa Lazzez, Mourad Zaied, Tarek M. Hamdani: Supervision, Data curation, Software, and Methodology. Ahmed T. Sadiq: Writing, Review & editing, Visualization, Formal analysis, and Investigation. Habib Chabchoub, Adel M. Alimi: Project administration, Conceptualization, Investigation, Validation, Supervision, and Review & editing.

Funding of Work

The manuscript did not require funding from any organization.

Clinical Trial

The manuscript does not contain any clinical trial.

Publish Declaration

Consent to Publish declaration: applicable

Consent to Participate Declaration

Consent to Participate declaration: not applicable

Ethical Approval

This manuscript does not pose any ethical concerns, as it does not utilize data of individuals' people but rather standard data.

REFERENCES

- [1] Mohammad Mustafa Taye." Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions". Published 25April 2023.
- [2] Divya P, Aiswarya VB Student." DEEP Learning: Methods And Applications". 2021 JETIR July 2021, Volume 8, Issue 7 www.jetir.org (ISSN-2349-5162).
- [3] Aditya Dubey, Akhtar Rasool." Recent Advances And Applications Of Deep Learning Method". Journal of Theoretical and Applied Information Technology 15th July 2022. Vol.100. No 13 © 2022 Little Lion Scientific.
- Augmented Reality and Ontology-based Approach for
Enhancing Arabic Text Recognition and VisualizationSsss Augmented Reality and
Ontology-based Approach for
Enhancing Arabic Text Recognition and
- [4] Ibomoiye Domor Mienye.Theo G. Swart." A Comprehensive Review of Deep Learning: Architectures, Recent Advances, and Applications". Published: 27 November 2024, Information 2024, 15(12), 755; <https://doi.org/10.3390/info15120755>.
- [5] Kamal Choudhary, Brian DeCost, Chi Chen, Anubhav Jain, Francesca Tavazza, Francesca Tavazza, Cheol Woo Park, Alok Choudhary, Ankit Agrawal, Simon J. L. Billinge, Elizabeth Holm, Shyue Ping Ong, and Chris Wolverton." Recent advances and applications of deep learning methods in materials science". Article number: 59 (2022).
- [6] Mohamm adreza Iman1, Khaled Rasheed and Hamid Reza Arabnia. " A Review of Deep Transfer Learning and Recent Advancements". Springer Nature 2021 LATEX template.
- [7] Prasad Kulkarni, Suyash Karwande, Rhucha Keskar, Prashant Kale, and Sumitra Iyer, " Fake News Detection using Machine Learning". ITM Web of Conferences 40, 03003 (2021).

- [8] Mohammed E.Almandouh, Mohammed F. Alrahmawy, Mohamed Eisa, Mohamed Elhoseny& A. S. Tolba. " Ensemble based high performance deep learning models for fake news detection". Published: 04 November 2024, Article number: 26591.
- [9] I. Kadek Sastrawan, I.P.A. Bayupati, Dewa Made Sri Arsa." Detection of fake news using deep learning CNN–RNN based methods". Received 3 July 2021; received in revised form 15 September 2021; accepted 6 October 2021 Available online 22 October 2021.
- [10] Azka Kishwara, Adeel Zafara. " Fake News Detection on Pakistani News Using Machine Learning and Deep Learning". April 15, 2022.
- [11] Sangita M. Jaybhaye, Vivek Badade, Aryan Dodke, Apoorva Holkar, and Priyanka Lokhande. " Fake News Detection using LSTM based deep learning approach ". ITM Web of Conferences 56, 03005 (2023).
- [12] Sheng How Kong, Li Mei Tan, Keng Hoon Gan, Nur Hana Samsudin." Fake News Detection using Deep Learning". 2020.
- [13] Kushal Agarwalla, Shubham Nandan, Varun Anil Nair, D. Deva Hema." Fake News Detection using Machine Learning and Natural Language Processing". Published: 10 November 2022. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume-7, Issue-6, March 2019.
- [14] Mohamad Abdolahi, Morteza Zahedi." A New Method for Sentence Vector Normalization Using Word2vec". Int. J. Nonlinear Anal. Appl. 10 (2019) No. 2, 87-96 ISSN: 2008-6822 (electronic) <https://dx.doi.org/10.22075/ijnaa.2019.4177>.
- [15] Khalid Shaker, Arwa Alqudsi." Approach for Detecting Arabic Fake News using Deep Learning". DOI: <https://doi.org/10.52866/ijcsm.2024.05.03.049> Received April 2024; Accepted June 2024; Available online August 2024.
- [16] Hanen Himdi, George Weir, Fatmah Assiri, Hassanin Al-Barhamtoshy." Arabic Fake News Detection Based on Textual Analysis". Arabian Journal for Science and Engineering (2022) 47:10453–10469.
- [17] Ahmed Redha Mahlous, Ali Al-Laith." Fake News Detection in Arabic Tweets during the COVID-19 Pandemic". (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 12, No. 6, 2021.
- [18] Ali Bou Nassif, Ashraf Elnagar, Omar Elgendy, Yaman Afadar." Arabic fake news detection based on deep contextualized embedding models". Neural Computing and Applications (2022) 34:16019–16032.
- [19] Bohan Li, Yutai Hou, Wanxiang Che." Data augmentation approaches in natural language processing: A survey". Available online 24 March 2022 <https://doi.org/10.1016/j.aiopen.2022.03.001>
- [20] Sebastien C. Wong; Adam Gatt; Victor Stamatescu; Mark D. McDonnell. " Understanding Data Augmentation for Classification: When to Warp?". Date Added to IEEE Xplore: 26 December 2016 DOI: 10.1109/DICTA.2016.7797091.
- [21] Sosuke Kobayashi." Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations". June 1 - 6, 2018.
- [22] Xiang Zhang Junbo Zhao Yann LeCun." Character-level Convolutional Networks for Text Classification". posted in Feb 2015 as arXiv:1502.01710.
- [23] Connor Shorten, Taghi M. Khoshgoftaar and Borko Furht." Text Data Augmentation for Deep Learning". Shorten et al. J Big Data (2021) 8:101, <https://doi.org/10.1186/s40537-021-00492-0>.
- [24] Heng Yang, Ke Li." Boosting Text Augmentation via Hybrid Instance Filtering Framework". Findings of the Association for Computational Linguistics: ACL 2023, pages 1652–1669 July 9-14, 2023 ©2023 Association for Computational Linguistics.