



The Detection of Fake Text News using a Dense-based 1D-CNN Deep Learning Algorithm

Khalid Abood Kamel and Jumana Waleed

Department of Computer Science, College of Science, University of Diyala

scicompms23@uodiyala.edu.iq

Received: 18 December 2022

Accepted: 20 March 2023

DOI: <https://dx.doi.org/10.24237/ASJ.02.02.728B>

Abstract

There are a lot of problems with fake news, which can make people think of things that aren't true. Social media is one of the fastest ways to get information out there because it has a big impact and can manipulate information in both good and bad ways. The goal of this paper is the optimal use of deep learning algorithms to solve the problem of the paper. The research problem is how accurately and to what extent can an individual distinguish between fake news articles using natural language processing and classification algorithms. What are the steps that can be taken to provide a solution? compared to the previous different methods to solve this problem, including some common deep-learning methods. In this paper, we can find fake news can be found by using the term inverse frequency document (TF-IDF) for feature extraction and a hybrid algorithm of One Dimensional-Convolutional Neural Network (1D-CNN) and Dense as the classifier. The experiments that the proposed dense-based 1D-CNN algorithm substantially outperforms other up-to-date related algorithms with an accuracy of 100%.

Keywords: CNN, NLP, TF-IDF, Fake news detection, text classification, News articles dataset.



الكشف عن الأخبار النصية المزيفة باستخدام خوارزمية التعلم العميق للشبكة العصبية التلافيفية أحادية البعد كثيفة

خالد عبود كامل و جمانة وليد صالح

قسم علوم الحاسوب – كلية العلوم – جامعة ديالى

الخلاصة

هناك الكثير من المشاكل مع الأخبار المزيفة، والتي يمكن أن تجعل الناس يفكرون في أشياء غير صحيحة. تعد وسائل التواصل الاجتماعي إحدى أسرع الطرق للحصول على المعلومات لأنها ذات تأثير كبير ويمكنها التلاعب بالمعلومات بطرق جيدة وسريعة. الهدف من هذه الورقة هو الاستخدام الأمثل لخوارزميات التعلم العميق لحل مشكلة الورقة. تتمثل مشكلة البحث في مدى دقة وإلى أي مدى يمكن للفرد التمييز بين المقالات الإخبارية المزيفة باستخدام معالجة اللغة الطبيعية وخوارزميات التصنيف. ما هي الخطوات التي يمكن اتخاذها لتقديم حل؟ مقارنة بالطرق المختلفة السابقة لحل هذه المشكلة، بما في ذلك بعض طرق التعلم العميق الشائعة. في هذا البحث، يمكننا العثور على أخبار مزيفة باستخدام مصطلح مستند التردد العكسي (TF-IDF) لاستخراج الميزات وخوارزمية هجينة لشبكة عصبية تلافيفية أحادية البعد (D-CNN1) وكثيف كمصنف. التجارب التي تفيد بأن خوارزمية D-CNN1 المقترحة كثيفة الأداء تتفوق بشكل كبير على الخوارزميات الأخرى ذات الصلة بدقة تصل إلى 100٪.

الكلمات المفتاحية: CNN، NLP، TF-IDF، كشف الأخبار المزيفة، تصنيف النص، مجموعة بيانات المقالات الإخبارية.

Introduction

People and governments talk about the spread of fake news a lot these days because it has a big effect on how people think and feel. People are trying to figure out what "fake news" means so that it doesn't hurt society. Fake news is made-up information that looks like real news but has a completely different purpose. So, it's hard for people to tell the difference between real and fake news on social media or in news agencies [1]. Spreading fake news leads to mistrust. Which leads to biased reviews that lead to biased perceptions. Fake news hurt people's minds, which makes them read less trustworthy news. This kind of false news is spread by people who want to make money from the number of views on their page or by giving a biased opinion to trick people into making a different choice, like during an election [2]. Fake news spreads faster



than real news because it is more interesting to the viewer. It's hard to tell the difference between real and fake news. People tend to for think that if they know their facts are correct, they will have biased opinions about the information. This makes them "yellow journalist. This shows information that a person will think about based on what he or she already knows and that will support their biased views and opinions. With the help of social networking sites like Facebook, which can spread rumors. One of the most popular areas of study in Natural Language Processing (NLP) is text classification. Text processing is used in almost every field, such as question-answering, e-commerce, entertainment, news media, the stock market, hotel management, healthcare, and more [3]. Feature extraction using TF-IDF and classification using a hybrid algorithm of 1D-CNN and Dense. In contrast to previous work, this algorithm has shown effective performance using the fake news dataset by providing excellent first-rate ratios, recall, accuracy, and accuracy.

Related Work

Several studies have been reported on the development of fake news detection. Several deep learning models have been used for rating optimization problems in fake news. Most of them are listed below.

Mohammadreza Samadi et al. [1], 2021, proposed three classifiers with different models that had already been trained to use news articles as inputs. Connect a Single Layer Perceptron (SLP), Multi-Layer Perceptron (MLP), and Convolutional Neural Network (CNN) after an embedding layer of new pre-trained models like BERT, RoBERTa, GPT2, and Funnel Transformer so that they can use the deep contextual representation that these models give them in addition to deep neural classifications. They evaluated the proposed models on three known fake news datasets: LIAR, ISOT, and COVID-19. The results of these three datasets show that the proposed models are superior at detecting false news compared to modern models.

Sastrawan et al. [4], 2021, provided research on a deep learning approach employing four distinct datasets and multiple architectures, including CNN, Bidirectional LSTM (Long Short-Term Memory), and ResNet (Residual Network). The initial dataset was the ISOT fake news dataset, followed by the Fake News Dataset, the Fake or Real News Dataset, and finally the



Fake News Detection Dataset. On all the datasets examined, the Bidirectional LSTM architecture beat ResNet and CNN. The accuracy percentages achieved were 99.24 percent, 98.99 percent, and 98.24 percent, respectively.

Nasir et al. [5], 2021, integrated both convolutional and recurrent neural networks into a new type of deep learning model for identifying fake news. The FA-KES 5 dataset was based on two fake news data sets. It had 804 news stories about the Syrian war (ISO and FA-KES). The number of real articles is 426, while the number of fake articles is 376. This makes for a well-balanced dataset (53 percent true versus 47 percent fake articles). On the other hand, the ISOT 6 dataset had 45,000 news stories that were almost equally split between true and false. In the proposed model, CNN was used to pull out local features, and LSTM was used to learn about long-term relationships. First, a CNN layer (one-dimensional CNN) is used to process the input vectors and pull out the local features that live at the next level. The output of the CNN layer is sent to the next layer, which is made up of LSTM and cells and is called the Recurrent Neural Network (RNN). CNN tried out the RNN layer, which learns how long-term relationships between local parts of news stories help decide if they are true or not. Experiments with two different real-world fake news datasets (100 percent accuracy on the ISOT dataset, which has 45,000 articles, and 60 percent accuracy on the FA-KES dataset, which has 804 articles) show that hybrid baseline approaches work much better than non-hybrid baseline methods. Last the amount of data used to train deep learning systems can help them.

Aslam et al. [6], 2021, suggested an ensemble-based deep learning model to categorize news as false or true utilizing the LIAR dataset. Two deep-learning models were utilized due to the nature of the dataset characteristics. For the textual attribute "statement," the Bi-LSTM-GRU-dense deep learning model was used. For the rest of the attributes, the dense deep learning model was used. The suggested study produced an accuracy of 0.898, recall of 0.916, precision of 0.913, and F-score of 0.914 utilizing simply the statement attribute, according to experimental data.

Azka Kishwar and Adeel Zafar [7], 2022, proposed to make the first complete dataset for spotting fake news in Pakistani news by using multiple valid news APIs. Multiple cutting-edge



artificial intelligence techniques have also been used to test the developed dataset. Five different types of machine learning were used: Naive Bayes, KNN, Logistic Regression, SVM, and Decision Trees. While CNN and LSTM were used in GloVe and BERT embeddings, they were not the only ones. All of the models and embeddings that were used were compared based on their accuracy, F1 score, accuracy, and recall. Based on the results, the LSTM that was set up with GloVe embeddings did the best. The study also looked at the misclassified samples by comparing them to how people would have rated them.

Linmei Hu et al. [8], 2022, gave a full review and analysis of DL-based fake news detection (FND) methods that focus on different things like news content, social context, and external knowledge. Methods that were supervised, poorly supervised, or not supervised at all were looked at. For each line, the representative looked at how to use different features in a planned way. Next, I talked about a few FND datasets that are often used and gave a quantitative analysis of how well DL-based FND methods work on these datasets. Finally, the remaining problems with current methods are looked at, and some promising directions for the future are pointed out.

Reham Jehada and Suhad A.Yousif. [9], 2022, They suggested that fake news be found. FNs using the multi-layer perception algorithm (MLP) as a class and the term inverted frequency document (TF-IDF) as a feature extraction method. A three-layer system has two stages: feed-forward and propagation-back (an input layer, one hidden layer, and an output layer). When our proposed algorithm was used to classify the FNs dataset, the accuracy was 95.47%.

Proposed System

This work has been done in five steps, which are shown in Figure 1. The next sections go into more detail about each of these stages:

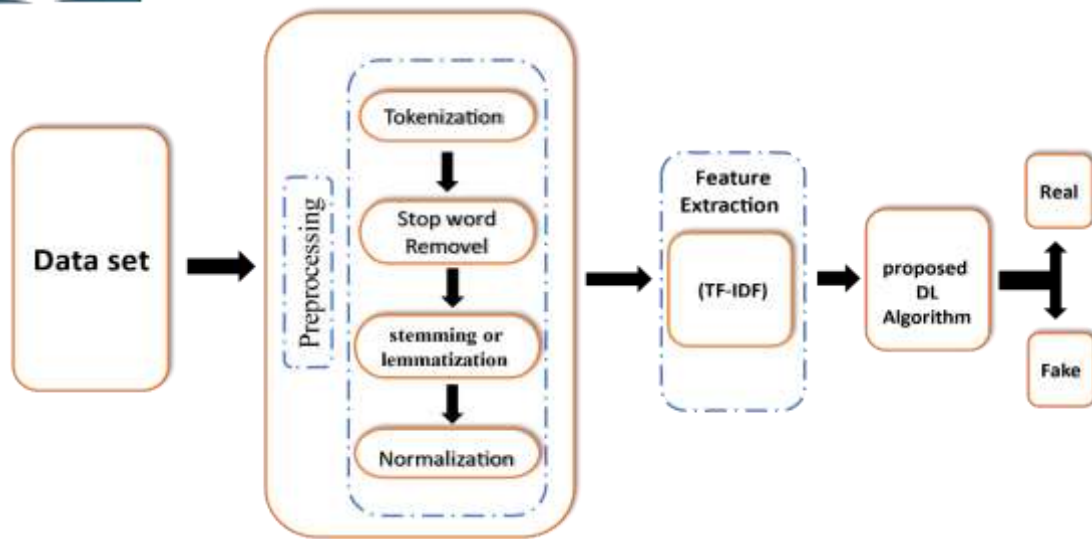


Figure 1: System work steps

1. Input Dataset of Real and Fake News

The dataset used in this work (the fake news articles.CSV file) was obtained from kaggle.com [10]. This dataset has about 45,000 records that were taken from different articles found on the internet (text, author, title, and label). After the missing rows were taken out of the dataset, it had 44,919 records, of which 21,417 were real news stories and 23,502 were fake news stories. The only thing that is used to spot fake news is the text. Figure 2 shows real and fake data segmentation. After performing the operations mentioned later the data set.

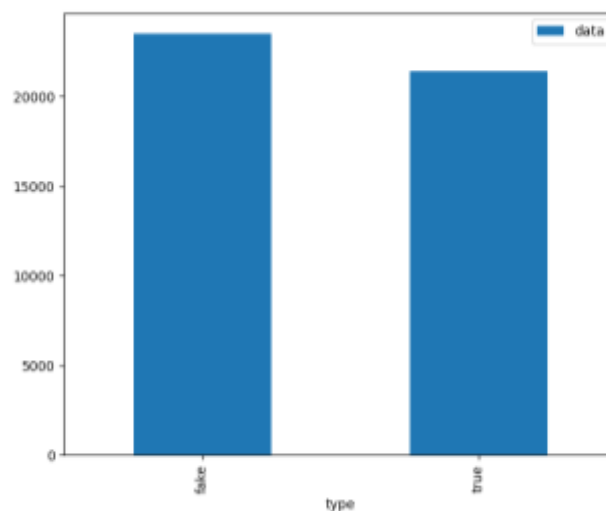


Figure 2: Data segmentation is real and fake.

2. Preprocessing Phase

After the data set is entered into the system, the text to be categorized is cleaned up and sorted as data part of pre-processing. The online text has a lot of noise and parts that do nothing, such as ads, scripts, and tags. Handling these words increases the dimensionality of the problem, which makes it harder to classify because each word in the text is being handled in one dimension (1-D). The data could be accurately preprocessed by removing noise from the text. This would speed up the classification process and improve the performance of the classifier, leading to a useful dataset. This stage is illustrated in Figure 3.

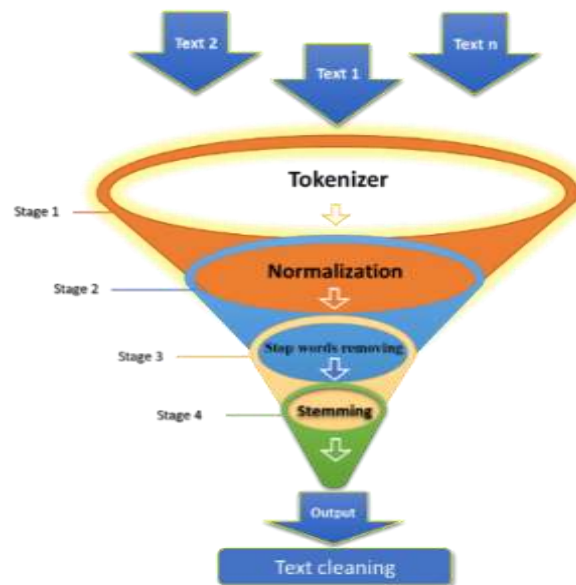


Figure 3: Stages of text cleaning

a) Tokenization

Tokenization means to break the sentence up into words. The words that were found after the tokenization process are called tokens. The first step in the text analytics problem was to make a list of tokens, which would be used later in another preprocessing step. The main reason to use tokenization is to find keywords that mean something. The problem with tokenization is that it's hard to tokenize a document that doesn't have any white space, special characters, or other marks.



The first step in most processing applications was to break up text into words. In the English language, words were separated by a blank space. So, for this language, which is called a "segmented language," it was easy to find the boundaries between tokens because most of them were already separated by punctuation and spaces. Each document has been broken up into symbols based on whitespace, as described in Algorithm 1.

b) Normalization

All of the document tokens were changed to either upper or lower case by this process since most reviews use both upper and lower case characters. So, tokens could be used to predict easily if they were put into a single format. In algorithm 1, all tokens' base roots were changed to lower case.

c) Stop word

Stop-words are English words that are used a lot but add little to what is being said. It's easy to say that it doesn't matter. "Stop words" are words that don't belong in a sentence. Before, these words were kept in a package called the corpus. There is more about corpus in the NLTK manual. It can be set up in the Python environment itself. The Natural Language Toolkit (NLTK) is a set of Python-based tools and programs for symbolic and statistical natural language processing (NLP) of the English language. Stop words are taken out of a sentence by breaking it into words and then checking to see if each word is on the NLTK list of stop words. If the word is already in the blog group, it is removed. Figure 4 shows what needs to be done to make the text look better.

d) Stemming:

By steaming, the words have been broken down to their base. Sometimes the stem item is not the same as the root, but this is still useful because most related words map to the same stem, even if the stem is not a valid root. Snowball stemming was used in algorithm 1 to send all tokens back to the base root. Stemming changes the words back to their original form (the root)



and cuts down on the number of word types or classes in the data. "Running," "Ran," and "Runner" will all be shortened to "run," for example, see Figure 4.

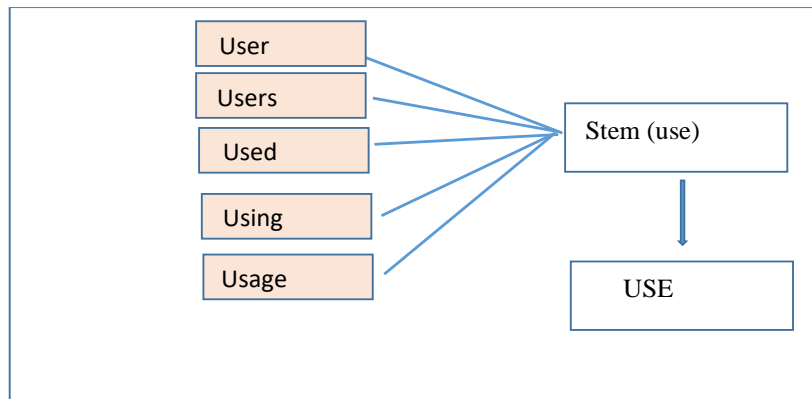


Figure 4: Process of stemming.

Algorithm 1: Preprocessing Algorithm
Input: Labeled Dataset
Output: Preprocessed Data
Step 1: each input document (D_i), where $I = 1, 2, 3, \dots, n$. Extract Word (EW_i) by dividing the document into words when reading in white space.
Step 2: For each word (EW_i), remove all stop words like numbers, symbols, and prepositions and return word (EW_i) without stop words.
Step 3: Remove all stop words and change all tokens to root.
Step 4: Change each token from all capital letters to all small letters.
End

3. Feature Extraction

Token extraction is a key step in news classification because it lets characteristics be taken from text news stories that have already been preprocessed. Using a vector space model, the following steps are taken to pull out the features:

The representation of news is news must be converted from its complete text representation to a news vector, which represents each pieces of news as an array of characteristics. The term "Frequency-Inverse Document Frequency". represents the weighting technique for the feature. It is a numerical statistic that shows how important a certain word is in a set of words. Most of the time, it is used to give weight to information in knowledge recapturing and data mining. Its



benefit is getting better based on how often a term shows up in the data, but this is cancelled out by how often the term shows up in the frame. That may have something to do with the fact that some words are used more often than others. This stemmer works well for removing suffixes from different sites, summarizing data, and putting it into different groups. It comes from two things called parameters (frequency and inverse document frequency). For better recognition, counts and saves the number of times each word appears in texts. The number of times a word appears in a certain set of data is the first statistic (TF). As in the equation (1).

$$TF(i, j) = \frac{\text{term } i \text{ frequency in document } j}{\text{total words in document } j} \dots\dots\dots (1)$$

The second statistic used to figure out how important a word is in a set of data is the Inverse Document Frequency (IDF). IDF has a process that gives less weight to words that appear more often in a group of text and more weight to words that appear less often. As in the equation (2).

$$IDF(i) = \log_2\left(\frac{\text{total documents}}{\text{documents with term } i}\right) \dots\dots\dots (2)$$

After that TF-IDF is counted for every term. As in the equation (3).

$$TF - IDF = TF(i, j) * IDF(i) \dots\dots\dots (3)$$

4. Classification using deep learning

Deep learning (DL) is a type of machine learning (ML), which is a artificial intelligence (AI) that is based on how the human brain works. It is a part of ML Cognitive Computation, which works like the biology of the human brain by taking data and processing it through neural networks. It is used in many areas, like finding fake news and sorting it into true and false news based on artificial intelligence models.

DL models can detect features that hide in texts that are not visible or undetectable by traditional readers and people. Whereas, in DL, the Convolutional Neural Network (CNN) is a leading DL tool that is commonly used in various sub-domains of the fake news detection system due to its ability to extract features and optimally categorize news.



5. Classification phase based on CNN

Convolutional neural networks are one of the main types of neural networks used to recognize text. It takes a text as input, processes it, and puts it into certain groups. Each text input will be sent through a series of Convolutional layers with filters (Kernels), pooling, and fully connected layers (FC) to classify an object with probabilistic values between 0 and 1. All one-, two-, and three-dimensional CNNs work the same way. The difference is in how the filter, also called a convolution kernel or feature detector, moves across the data and how the data is organized. In this thesis, the CNN layers and their parameters will be explained and shown in a single dimension.

The proposed 1D-CNN model is introduced and shown in Figure 5. in which the layers (12 layers) are defined:

- Five convolutional layers for feature extraction of type 1D.
- Five ReLU 1D layers
- One flattens layer. and as shown in the algorithm (2).
- **One layer that was fully connected, which was shown by the (Dense).**

Algorithm (2): 1D-CNN	
input text messages after token extraction	
Output the category class A OR B	
Begin	
1	Form text messages (Tm) Make the Quantization.
2	Convert the messages to an array of numbers.
3	Log 1D. CNN: Make use of network layers -Conv 1, with 1024 filter, AF = ReLU, Max_pool, with P=1 -Conv 2, with 512 filter, AF = ReLU, Max_pool, with P=1 -Conv 3, with 256 filter, AF = ReLU, Max_pool, with pool_size= 1 -Conv 4, with 128 filter, AF = ReLU, Max_pool, with P=1 -Conv 5, with 64 filter. Flatten = conv 5 layers Dense layer = Fully connected layer, AF = softmax
4	Classify the text message to which class it belongs.
5	Return to Step 4
6	repeat Step 7
END	

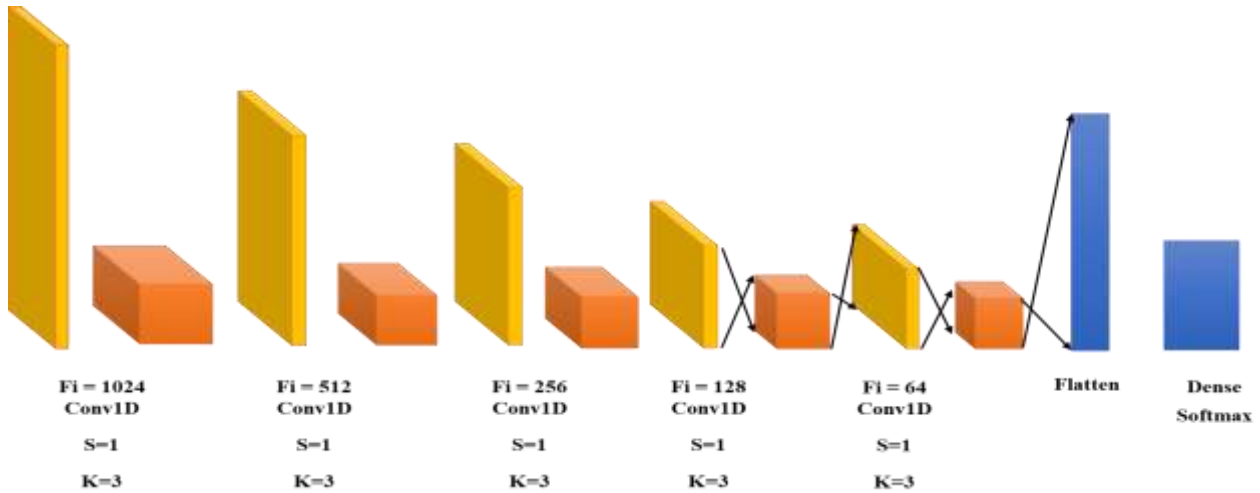


Figure 5: Layers model of 1D-CNN.

Where fi value means the size of the filter, K (Kernal) means the number of windows, and S (stride) means the number of steps within the implementation of the above operations.

Tables 1 and 2 illustrate the rating of the obtained results concerning the proposed system and the previous works, respectively.

Table 1: The obtained Precision, and Recall, and F1-score for the proposed system.

DL	Precision	recall	f1-score	accuracy
1D-CNN	100	100	100	100

Table 2: The obtained Precision, Recall, F1-score from previous research.

Author	Model	Precision %	Recall %	F1-score %	Accuracy %
Mohammadreza Samadi et al. [1], 2021	RoBERTa-CNN	98.30	96.27	97.27	97.43
Sastrawan et al. [4], 2021	CNN, Bidirectional LSTM, and ResNet	99.95	99.95	99.95	99.95
Nasir et al. [5], 2021	hybrid CNN-RNN	48	48	46	50
Aslam et al. [6], 2021	The Bi-LSTM-GRU-dense	91.3	91.6	91.4	89.8
Azka Kishwar and Adeel Zafar. [7], 2022	LSTM	95	94	95	94
Reham Jehada and Suhad A.Yousif. [9], 2022	multi-layer perception algorithm (MLP)	96.2	94.49	-	95.47
The proposed Model	1D-CNN	100	100	100	100

The Experiments were performed using a size of 100 epochs, 64 batches, a learning rate of 0.001, and an Adam optimizer. Figure 6 shows a comparison of the results of previous studies with the results of our system.

The results in the proposed Model were high due to the use of feature extraction which produced robust features consistent with the algorithm used.

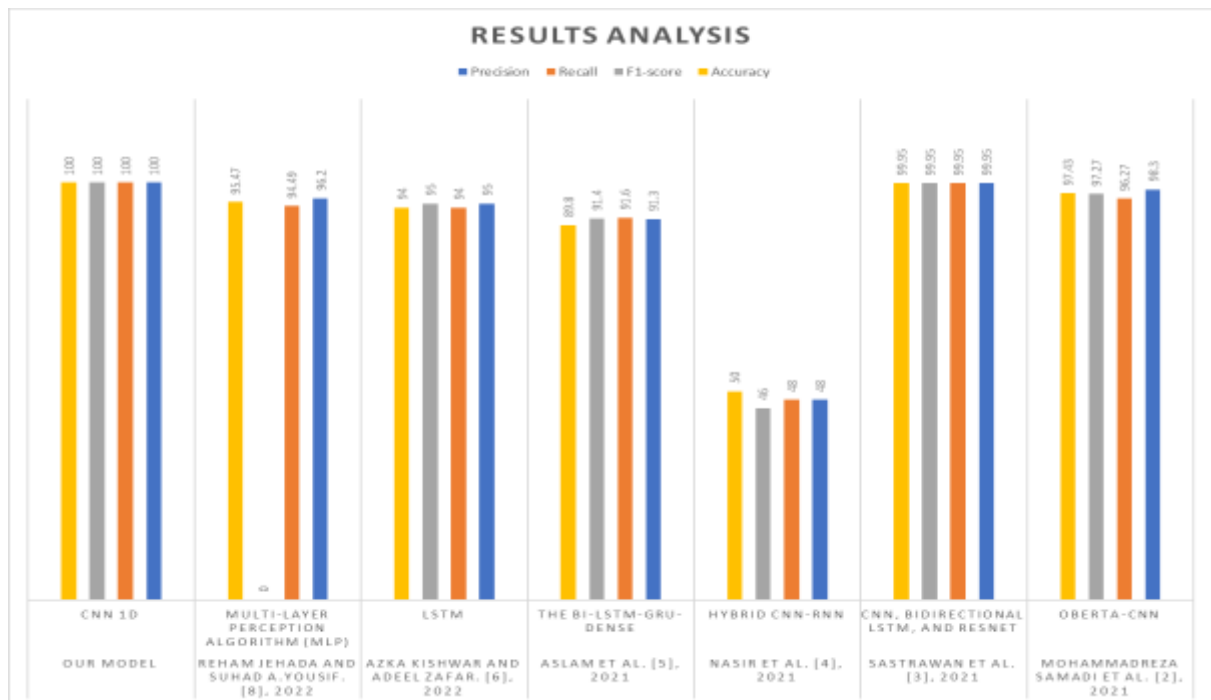


Figure 6:3 the achieved results of previous studies and the results of our study

6. Evaluation Metrics and Results

In this paper, the evaluation metrics (accuracy, precision, and recall) are defined in the equations (4, 5, 6 and 7) as follows [11]:

Accuracy is the number of times fake news and real news were labeled correctly out of all the times they were labeled correctly and wrongly. Equation (4) shows how accurate something is.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots (4)$$



Precision, which is the number of times fake news was correctly identified as fake out of all fake news (good news), is shown in Equation (5).

$$\text{Precision} = \frac{TP}{FP+TP}. \quad \dots\dots\dots (5)$$

Recall, which is the model's sensitivity, is how well the model can find good examples of fake news, which is defined by the equation (6).

$$\text{Recall} = \frac{TP}{FN+TP}. \quad \dots\dots\dots (6)$$

The F-measure can be calculated by taking the weighted average of the recall and precision measures, as shown in Equation (7).

$$F - \text{Measure} = \frac{(2 \times \text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}. \quad \dots\dots\dots (7)$$

Conclusion

Recently, with the popularity of social media platforms, fake news (news that includes fake information) has become widespread and due to the negative effects of spreading fake news in various fields, the automatic detection of fake news has received more attention in the past few years, especially with the advent of detection systems based on deep learning. A deep learning-based fake news detection algorithm is proposed in this paper. Here, fake news can be detected using two main stages Feature extraction using TF-IDF and classification using a hybrid algorithm of 1D-CNN and Dense. Dissimilar from previous work, this algorithm has shown effective performance using the fake news dataset by providing excellent percentages of f1-score, recall, precision, and accuracy. In future work, for enhancing the proposed algorithm, we seek to detect counterfeit news in texts using various languages. which represents one of the crucial challenges for solving the detection of multi-lingual counterfeit news. The research problem is how accurately and to what extent can an individual distinguish between fake news articles using natural language processing and classification algorithms. What are the steps that can be taken to provide a solution? a detection model is presented to classify fake news with



the effect of TF-IDF on the dataset. Using deep learning algorithm (CNN1D). using the dataset (the fake news articles.CSV file) from kaggle.com is used in this thesis. the model achieved maximum accuracy (100%). The results achieved are better than the related inline works, so the use of this algorithm enhances the classification accuracy. Where these results were achieved because of the processes used, which we mention with their benefits:

- It was necessary to remove the empty rows, remove the columns (Author, Title, Label) and keep only the (Text) column of the data set used during the data pre-processing steps to improve training and execution time.

- Pre-processing steps using our models (stopwords, punctuation, capital letters, whitespace, special characters, and word recovery) are necessary because they had a significant impact on increasing classification accuracy.

- Inverse Frequency Documentation Method (TF-IDF). Necessary to reduce overfitting during model training, which saved training time on the (CNN1D) model, bringing the accuracy to a maximum of 100%. Through work we have found that this method works well with the (CNN1D) model because it increases classification accuracy, which is sufficient to prevent overfitting and increase accuracy.

- Mohsen Adam used with the functions of this data set has an important role in improving classification accuracy because of its benefit in updating weights.

References

1. M. Samadi, M. Mousavian, S. Momtazi, Deep contextualized text representation and learning for fake news detection, *Inf. Process. Manag.*, 58(6),10272392021)
2. S. B. S. Mugdha, S. M. Ferdous, A. Fahmin, Evaluating Machine Learning Algorithms for Bengali Fake News Detection, *ICCIT 2020 - 23rd Int. Conf. Comput. Inf. Technol. Proc.*, no. December 2020, 2020
3. P. Kumar Roy, A. Kumar, Convolutional Neural Network for Text: A Stepwise Working Guidance, *SSRN Electron. J.*, (2021)



4. I. K. Sastrawan, I. P. A. Bayupati, D. M. S. Arsa, Detection of fake news using deep learning CNN–RNN based methods, ICT Express, 2021
5. J. A. Nasir, O. S. Khan, I. Varlamis, Fake news detection: A hybrid CNN-RNN based deep learning approach, Int. J. Inf. Manag. Data Insights, 1(1), 100007(2021)
6. N. Aslam, I. Ullah Khan, F. S. Alotaibi, L. A. Aldaej, A. K. Aldubaikil, Fake Detect: A Deep Learning Ensemble Model for Fake News Detection, Complexity, 2021
7. A. Kishwar, A. Zafar, Fake news detection on Pakistani news using machine learning and deep learning, Expert Syst. Appl., 211(April 2022), 118558(2023)
8. L. Hu, S. Wei, Z. Zhao, B. Wu, Deep learning for fake news detection: A comprehensive survey, AI Open, 3(September), 133–155(2022)
9. R. Jehad, S. Yousif, Classification of fake news using multi-layer percepti on, AIP Conf. Proc., 2334(March), (2021)
10. Data set: <https://www.kaggle.com/datasets/jainpooja/fake-news-detection>
11. S. A. Yousif, Z. N. Sultani, V. W. Samawi, Utilizing arabic wordnet relations in arabic text classification: New feature selection methods, IAENG Int. J. Comput. Sci., 46(4), 1–12(2019)