Available online at www.garmian.edu.krd

Passer Journal

Passer 5 (Issue 2) (2023) 262-271

http://passer.garmian.edu.krd/

Kurdish Fake News Detection Based on Machine Learning Approaches

Dana Abubakr Salh[1*], Rebwar Mala Nabi[1]

[1]*Department of Information Technology, Technical College of Informatics, Sulaimani Polytechnic University, Sulaimani, Kurdistan Region, Iraq.*

Received 08 January 2023; revised 26 April 2023;
accepted 27 April 2023; available online 25 May 2023

DOI: 10.24271/PSR.2023.380132.1226

ABSTRACT

The widespread use of social media platforms and the internet has increased information sharing, including both true and false news. Detecting fake news is challenging, and several studies have been conducted to automate this process for popular languages such as English and Arabic. However, more research must be done on detecting fake news in low-resource languages such as Kurdish. This gap was addressed, and a publicly available Kurdish fake news dataset (KDFND) was used, comprising 100962 news articles, among which 50751 are real, and 50211 are fake news labeled as Real and Fake. In this study, three techniques were implemented to extract features from news texts, including word embedding, term frequency-inverse document frequency, and count vector, and three various machine learning and deep learning classifiers were used (Random Forest, Support Vector Machine, and Convolutional Neural Networks) to identify the fake news dataset. The results showed that fake news with textual content could be identified, especially when convolutional neural networks are used. According to the experimental results of the study, CNN performs better than the other models, with an F1-score of 95% and an accuracy of more than 91% percent. These findings indicate that machine learning methods can efficiently detect fake news in low-resource languages like Kurdish, even in complex environments.

Keywords: Fake News, Kurdish Fake News, Text Classification, Kurdish News, Natural Language Processing., KLPT, Machine learning, Deep learning.

1. Introduction

The world lives in an era of vast information, and the amount of data produced by social networking sites (SNSs) and social media is increasing quickly. SNSs are electronic platforms where countless users communicate daily in digital text format. Also, the development of social media platforms has made publishing and sharing information much simpler and quicker. Individuals could now access global information with only one click[1]. These platforms provide various services, including publication and content creation, to hundreds of millions of users. A significant source of news that many people read daily is social media. One primary social media is Facebook, a popular news-sharing platform[2]. Unfortunately, not every news story that Facebook publishes is reliable and trustworthy. Many people attempt to spread false and inaccurate information to influence public opinion[3]. Fake news is usually generated for financial and political gain, such as to manipulate the stock market or influence elections for the president. It might have a significant impact on society. For instance, fake news contributed significantly to Donald Trump's victory in the 2016 US presidential election, as demonstrated by the 2016 US presidential election and the Brexit referendum events[4]. Data and text mining techniques are used in social media mining in conjunction with social network analysis and information retrieval to uncover hidden data valuable for propagating news, sentiments, and opinions via social media platforms. Essentially, SA analyzes natural language texts to identify characteristics that best describe an item, like a person, service, or organization. One of the main SA applications that aim to detect misleading or fraudulent information on social media is called "fake or false information." Fake news has been described in many different ways up to this point, and there are still numerous misunderstandings[5]. Depending on the objective and goal of the definer, it can be determined what the term means. For instance, it is referred to as "news articles that are presented as factual yet have no factual basis." "Fake news, or hoax news," according to[6], "refers to information or propaganda that has been intentionally fabricated and disseminated under the appearance of authenticity."

* Corresponding author
E-mail address: dana.abubakr.salh@spu.edu.iq (Instructor).
Peer-reviewed under the responsibility of the University of Garmian.

In addition, detecting fake news on social media can be a tedious task because it can be challenging to identify it on social media for several reasons, including the following: (1) It takes a lot of time to validate suspicious news and search for verified evidence;

(2) There are few trustworthy sources; and (3) The posts on social media are typically very noisy and lack the necessary information to be easily verified as true[7]. However, a thorough understanding of fake news and its sources is necessary to begin tackling this issue. Only after that can one begin to investigate the many approaches and disciplines of machine learning (ML), natural language processing (NLP), and artificial intelligence (AI) that may assist in combating this issue[8].

On the other hand, Kurdish is an Indo-European language spoken primarily in northern Iraq and Syria, as well as in central and eastern Turkey and western Iran[9]. According to some variant reports, there are different dialects of the Kurdish language due to the uncertain number of populations that speak this language, ranging from 19 million to 28 million. In addition, unlike English, the Kurdish language is written from right to left[10]. Furthermore, the most popular dialects of this language are Kurmanji (Northern Kurdish), Sorani (Central Kurdish), Southern Kurdish, Zazaki, and Gorani[9]. The main reasons that Kurmanji and Sorani, the two most common dialects, have drawn attention to NLP research are their different grammar, vocabulary, and mutual phonetic similarity. Kurmanji is a language spoken by a large population in Turkey, Syria, Iraq, Iran, Armenia, and Lebanon. Sorani is a second dialect commonly used and mainly spoken by Kurds in Iran and Iraq[10].

Although the ability to evaluate the reliability of postings written in Kurdish is still in its infancy, numerous studies have been carried out to identify articles in English and Arabic as fake news. Construct a machine learning model in this study that measures the integrity of Kurdish news to eliminate and remove false information. The fake Kurdish dataset has been evaluated using multiple machine learning and deep learning models, including Support Vector Machine (SVM), Random Forest (RF), and Convolutional Neural Network (CNN), and the results from these models have been compared.

2. Problem Statement

We need to automatically identify fake news using machine learning and deep learning techniques based on features collected from the news content and information about the publishing user to reduce the consequences of publishing fake Kurdish news. In other words, the objective is to create a binary classifier that can identify a particular news item as "real" or "fake" and offer a probability estimate. There have been numerous study efforts to identify fake news in social media from other languages, but very few have been performed to identify fake news in Kurdish. However, developing a machine learning and deep learning model for detecting fake Kurdish news proved problematic for several reasons. (1) Most NLP tools do not support the Kurdish language. (2) The noisy nature of news content.

The structure of this article is as follows: The issue statement in the area of fake news identification is described in Section 2. The related works on fake news detection are examined in Section 3. The proposed methodology is explained in depth in Section 4, and the research findings are presented in Section 5. The benchmarking and comparative study is presented in section 6. The comprehensive investigation has been concluded, and the future trends are presented in Section 7.

3. Related Work

The problem of "detecting fake news on social media" has been thoroughly studied, and numerous models have been created to aid in solving it. Natural language processing methods and several machine learning or deep learning models were employed. A brief description of potential works in this field is included in this section.

Authors[2] investigated the Random Forest (RF) classifier on a Fake News Challenge dataset compiled from Facebook posts from reputable sources only by the development of a model utilizing a count vectorizer (using word tallies) or a TFIDF matrix (term frequency-inverse document frequency). The cross-validated accuracy score for this model is 91.7%, the true positive score is 92.6%, and the AUC score is 95%.

Thaher H. et al., 2021[5]. The researchers developed a well-performing classification model for the intelligent detection of fake news in Arabic tweets. As a result, this paper introduces a hybrid artificial intelligence model for identifying the credibility of Arabic tweets. For this, they investigated the efficiency of most text vectorization models (TF, BTF, and TF-IDF) with several machine learning algorithms (KNN, RF, SVM, NB, LR, LDA, DT, and XGboost) to identify the efficient model that fits the tested data. The performance of the TF model is better than other models only with the XGBoost classifier, with accuracy equal to 0.7781 and the F1_score equal to 0.8042. Finally, a wrapper FS method utilizing HHO and other meta-heuristics was applied to filter out irrelevant features and enhance the adopted classifier's performance.

In this study[8]. The dataset was collected from online social forums like Facebook and Twitter. It included news reports from various domains to protect most of the news rather than specifically classifying world news. The authors used an artificial intelligence approach for fully automated news classifiers. The optimal pre-processing and CNN classification combination were discovered through more than a hundred tests. As a result, the best ML classification technique for identifying optimistic fake news, hybrid SVM, is adopted. It used a TF and TF-IDF that are well stated in the NLP model. The suggested model achieved a maximum accuracy of 91.23% using unigram features and a hybrid SVM classifier.

R. Azad et al. (2021)[11]. The first experience of detecting fake news in the Kurdish language was tried. The authors prepared two sets of news. The first set includes (5000) real and (5000) fake news from Facebook pages and Sorani Kurdish websites, and the second set contains (5000) same real news and (5000) fake news that scraped from real news is considered such fake news in this set. In this study, in the investigation steps, five machine learning classification methods were used, including Logistic Regression (LR), Nave Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM), and Random Forest (RF). The evaluation results show that the SVM classifier outperformed the highest accuracy with 88.71%, while Random Forest scored a 79.08% accuracy rate for only one set of manipulated text data. Furthermore, the study did not label the data based on the Kurdistan Journal

Syndicate rules and regulations. Last, only a few features had been extracted from the dataset.

This paper (2020)[12] presents data on articles that detect automatic fake news in Urdu. The dataset contains five domains (health, sports, business, showbiz, and technology), separated into training and testing parts. Also, there was a slightly imbalanced ratio of approximately 60% real news and 40% fake news in news articles labeled with "fake" and "real" classes. In this study, different Natural Language Processing (NLP) domains were tried to detect the propagation of fake news on the Web. Also, this paper used two traditional ensemble machine learning methods, LSTM and Transformers (BERT), in neural network-based solutions. The investigation results indicated that logistic regression achieved the highest F1-macro score of 0.90%.

Researchers in 2019[7] explored that fake news significantly impacts politics, especially like a domino effect on the people, as happened in the last US presidential election. So, a dataset used as a source is named the LIAR dataset. This study proposed a new model for tuning the information gained from the data to make predictions. First, the data was pre-processed, and then Natural Language Processing (NLP) methods were used to create an intelligent detection model to indicate fake news in the LIAR dataset. After that, this model was used to extract some features from TF-IDF word n-gram features to be classified with machine learning algorithms. When feature extraction was finished, the model was trained and tested. The evaluation results showed that the performance of the logistic regression classifier was lower than the SVM performance on this dataset with that model.

In this study[3], the researchers used supervised classification models on the publicly available Arabic dataset to identify fake news from Arabic tweets using machine learning. For this, they built different features and divided them into two groups: features based on content and users. Four machine learning algorithms are used during the learning process, including DT, RF, AB, and LR The logistic regression, which has had its model optimized, achieves the highest results in terms of recall (83%). In contrast, the random forest, which depends on model optimization, produces the best results in accuracy (76%).

Nowadays, the spread of false information is a significant problem. The rapid development and expansion of social media platforms have filled the information exchange gap in daily life. Social media is the primary channel for disinformation to spread at an exponentially high rate. Given the creation and dissemination of fake news, automatic classification and detection of such fake news stories are urgently needed. It takes a lot of time and energy to identify false news manually. Therefore, much research has been conducted to automate fake news detection in popular languages such as English, Arabic, and Persian. Whereas there are a few studies on detecting fake news in low-resource languages like Kurdish, On the other hand, after conducting an intensive investigation, only one study was found regarding the creation of fake news detection for Kurdish. Table 1 represents related works on Kurdish, Arabic, and right-to-left languages.

Table 1: list some related works.

Reference	Language	Dataset	Classifiers	Features
[11]	Kurdish	FakeKurdNews	NB, DT, RF, LR, SVM	TF-IDF
[13]	Arabic	Collected 206,080 tweets	NB, KNN, RF, RC, J48, Logistic	TF-IDF
[14]	Arabic	It contained 100 news with hundred comments	KNN, DT, NB, SVM	TF-IDF
[3]	Arabic	1862 Arabic Tweets	RF, DT, LR, Ada	User-related feature
[15]	Arabic	Collected 3235 records	LR, SVM, RF, NB, SGD, NN, DT	BOW, TF-IDF, BOW+SF, TF-IDF+SF
[16]	Arabic	3042 news articles	LSTM, CNN-LSTM	Word-Embedding, Word 2vec
[17]	Persian	738 Persian rumor tweets	DT, SVM, SMO, NB	Content-based feature
[18]	Persian	14020 Persian tweets	LSTM	Word 2vec
[19]	Persian	2124 news articles	SVM, RF, LG, NB, stackLSTM, Majoruty	Word embedding
[20]	Persian	500 Persian tweets about Covid-19	SVM, DT, NB, MLP, LR	TF-IDF

4. Methods and Materials

This section shows the methods and techniques used in this work to find fake news in the chosen dataset. Additionally, Section 4.1 describes the datasets and other relevant information. A brief explanation of NLP is provided in Section 4.2. The data pre-processing is addressed in Section 4.2, feature extraction techniques are discussed in Section 4.3, the ratio of split data is displayed in Section 4.5, and the methods and algorithms required to solve this problem are provided in Section 4.6. The system architecture for the stages of our approach is depicted in Figure1.

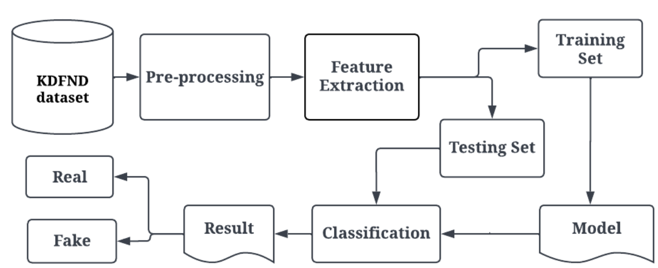

Figure 1: Fake news detection model.

4.1 Dataset Description and Architecture

Classifying a piece of news as "false" could be very hard and time-consuming. As a result, a dataset previously collected and classified as fake news was generated. The KDFND dataset is the data source used for this research. A short description of the data files used for this study is provided below. The dataset, initially collected by[21] and openly available in "Mendeley Data," has been used in this study to classify fake news. The dataset includes a collection of news gathered from several resources on the Web, including famous Kurdish news websites that the Kurdistan Journal Syndicate officially recognizes. Also, Facebook pages are the primary data collection source for fake and real news. The public websites cover three separate Kurdistan cities. The Kurdistan Regional Government of Iraq's three cities are Erbil, Sulaimani, and Halabja. The news articles are written in Kurdish. This dataset is also notable for being the first and largest in the Kurdish language to concentrate on the Sorani dialect. The dataset was collected from Jan. 01, 2021, until Dec. 31, 2021. The dataset utilized in this study includes 100962 articles categorized as true (real) and false (fake) news about economics, society, politics, entertainment, health, and sports; the total number of samples of both fake and real news is displayed in Table 2. The dataset also considers the post ID, text, URL, posting date, source, and label. Only the text and label variables from the dataset were used for this classification task to keep the process simple.

Table 2: The number of real and fake news in the KDFND dataset.

Sr. No.	Article Type	Frequency
1	Fake News Articles	50211
2	Real News Articles	50751

Furthermore, the KDFND dataset is currently the largest fake news dataset in the Kurdish language. Also, figure 2 is a pie chart showing the number and percentage of real and fake news samples. The dataset was analyzed, pre-processed, trained, and tested, and then three classification models were applied. Finally, experiments were performed on the test set. To our knowledge, this is the first study to use this dataset with machine learning and deep learning algorithms.

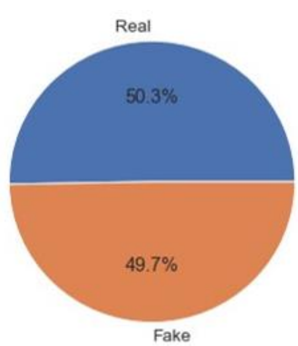

Figure 2: The percentage of real and fake news articles in the dataset.

4.2 Dataset statistic

We also performed some descriptive statistics on the dataset and provided statistical measures for the dataset. In order to see the different measures in the news article that describe text news, It is crucial to understand the length and other purposes so that one can detect the purpose of a news article and differentiate between different sizes of text news. Table 3 represents statistical measures of the corpus.

Table 3: Represent statistic measures of the corpus.

Stat. Measures	Count
Number of texts	100962
Max length of the text	2076
Min length of the text	1
The mean length of the text	23.17

Another form of visualization and descriptive analysis is the most frequent word. We will use a bar chart to explore the most frequent words in the corpus. The bar chart is interesting because, even with sorting, one can tell (with a little effort) by looking at the graph which word is most frequent and which is least frequent. The size of each bar represents the frequency or importance of each word. As can be observed from Figure 3, the generated text news of Facebook and website links are highly frequent in the corpus.

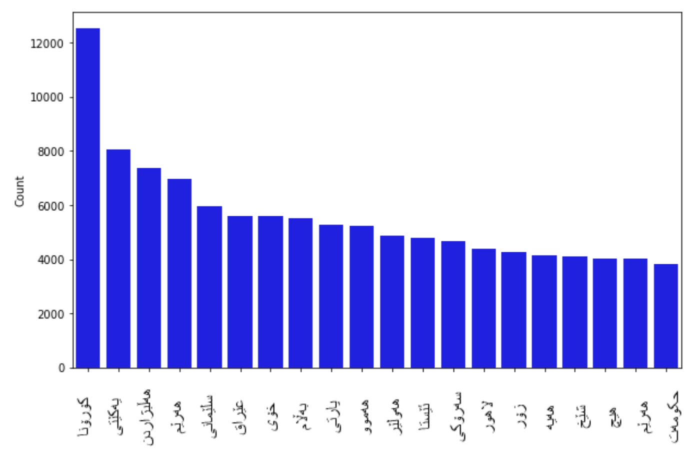

Figure 3: The 20 most frequently used words in the text news.

4.3 Natural Language Processing

As a subfield of artificial intelligence (AI), natural language processing (NLP) deals with processing and analyzing large volumes of natural language data, including speech and text. When it comes to text, for example, the main goal is to turn unstructured information into a structured form that can be analyzed to learn something useful. Different methods for processing and structuring sentiment data are included in NLP[22].

4.4 Pre-processing data

Before training, testing, and modeling, the data needs to be pre-processed. The real news and fake news are combined before proceeding to subsequent phases. We excluded columns from the datasets during the cleaning phase that were unnecessary for processing and improved data quality by checking for null or missing values[23]. Also, cleaning up the data text before feeding the classifiers in the pre-processing step is important. To accomplish this, unnecessary spaces, URLs, foreign language phrases, emojis, punctuation, and special characters like @, %, &. were deleted from the data[14]. Additionally, the library KLPT (Kurdish Language Processing Toolkit) developed by[24] has been imported to provide for removing stop-words in the Kurdish language " دە، لەگەڵ، بێ، بێجگە، بەبێ، دەکات، لەو، هەروەها، کە، و، نێوان، دوای، جگە، لە، لەبەینی، لەبەر، وەک، پاش."

Later, using a list of common prefixes and suffixes, the library's KLPT-Stem was imported to stem and transform the words to their base or origin form. The text should be divided into words using KLPT-tokenize, the final step. Tokenization is the process at this stage. The cleaned, attractive dataset was now prepared for the feature extraction stage. Figure 4 depicts the phases of data pre-processing.

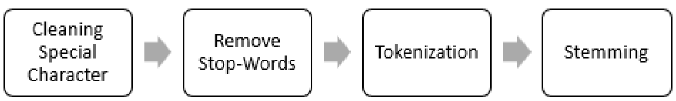

Figure 4: Pre-processing Steps.

4.4.1 Data cleaning

One of the major tasks of natural language processing (NLP) is data cleaning. We have removed from the collected data the special characters such as f_; @; %; &. Also, we have removed URL links, words in foreign languages, and all types of emojis[6]. For example, changing " بەرشلۆنە دەچێتە یۆرۆپالیگ! 🤔 https:/..." to " بەرشلۆنە دەچێتە یۆرۆپالیگ ".

4.4.2 Remove Stop-Words

Some Kurdish terms that are not essential in English sentences must be removed to improve the environment for finding information. Because these words are common in Kurdish Sorani texts, they increase the noise in the results. In this study, we removed stop-words in text from the dataset[25]. For example, after removing stop-words, " [کوردەکان, چۆن, کران, بە, نەوەی, جنۆکە]" is obtained as " [کوردەکان, چۆن, کران, نەوەی, جنۆکە]".

4.4.3 Tokenization

The tokenization process turns text into symbols and words separated by white space or commas. In addition, Tokens are a basic meaningful unit of a sentence or a document[16]. In this study, we are splitting text data by comma like" [بەرشلۆنە, دەچێتە, یۆرۆپالیگ]."

4.4.4 Stemming

Stemming is a technique to remove the suffixes and prefixes from words and end up with a word stem. We can reduce or aggregate several words into a common base form using stemming. In addition, returning a word to its root form or origin improves text extraction performance[26]. It also helps classifiers have fewer words as features. For instance, " [کوردەکان, چۆن, کران, نەوەی, جنۆکە]" Then, for each word, take back the root " کورد, چۆن, کران, نەوە, جنۆکە".

4.5 Feature Extraction

The challenge of processing items with high dimensions exists in text categorization. Documents contain a variety of terminologies, lexical patterns, and word groups that increase the computational complexity of learning[27]. Text vectorization, or feature extraction, is a problem-solving method that involves turning unstructured text into structured representation (numerical features) so that machine learning and deep learning algorithms can be used for knowledge extraction and mining. In simple terms, word-based statistical measurements are used to extract the numerical features[28]. Extraction of useful features from the actual news content is, therefore, a challenging task because fake news spreaders could make the content of the fake news look like real news. Hence, implementing a feature reduction for reducing the size of text features and prohibiting features of huge dimensions is the ideal way, according to[29]. In our work, different techniques are utilized in this model to score the word occurrences. The popular methods include count vectorizers, TF-IDF, and word embedding. The three strategies are utilized to extract useful features from news content to determine the most appropriate text representation model from the ones listed above, specifically for detecting fake news in Kurdish. Various classifiers are then fed the extracted features because machine learning and deep learning algorithms cannot comprehend the original text. Each extracted feature was utilized in one of the classifiers from the SciKit Learn Python package, described in the section below.

4.5.1 Count Vectors

Count the Vectorizer is a great Python algorithm that turns text into a vector based on how many times each word appears in the document. This tool is useful when many texts can be converted into vectors[28]. It is a notational representation in the form of a matrix data set in which each row represents the corpus, each column represents a corpus term, and each cell represents the count (frequency) of a specific document[30]. A count vector can tokenize the entire dataset and calculate the occurrence frequency of these tokens in each calculated document using Python's Scikit-learn package[29].

4.5.2 TF-IDF

It stands for Frequency-Inverted Document Frequency (TF-IDF), the most common method for extracting linguistic features in natural language processing. It detects fake news by looking for specific and suspicious tokens in news content[31]. This technique weights the tokens' frequency according to the document's importance. The TF-IDF method is used with document pre-processing pipelines using the Scikit-Learn library. The product of term frequency (TF) is evaluated in this approach by measuring each topic occurrence within a document and weighting it by its importance value (IDF). Equation (1) gives the formula for calculating the TF. As a result, TF represents the term frequency of each topic, and IDF represents the critical measurement of the issue, which results in a weight matrix for each case in the dataset. The indicated TF-IDF values could be calculated using the equation below. (2)[32] displays the IDF formula.

$$\mathrm{TF(t,d)} = \frac{\text{Number of times t occurs in a document 'd'}}{\text{Total word count of document 'd'}} \quad (1)$$

$$\mathrm{IDF(t)} = \log_e\left(\frac{\text{Total number of documents}}{\text{Number of documents with term t in it}}\right) \quad (2)$$

The TF-IDF should then be ascertained as the next step. The inverse document frequency combined with the term frequency is known as the TF-IDF[33]. Which formula is displayed in (3).

$$\mathrm{TF-IDF(t,d)} = \mathrm{TF(t,d)} * \mathrm{IDF(t)} \quad (3)$$

TF-IDF vectors can be generated using different input tokens, such as characters, words, and n-grams.

4.5.3 Word-Embedding

The word "embedding" can represent features in traditional classifiers or as an initialization step in deep learning neural networks. Word embedding works well with large corpora because it can display real-valued texts by embedding all semantic and syntactic meanings. Notably, it is one of the key advances in the performance of deep learning techniques in challenging natural language processing[34]. It converts the words or phrases into a real-valued vector representing the relationship between words and their semantic distance. We use a publicly pre-trained word vector mode to accomplish word embedding in this architecture. They are neural networks that are used to reconstruct linguistic concepts. As an input, this method takes a large portion of the text and generates a vector space with more than a hundred dimensions, but each word in the text body is assigned to a corresponding vector in the space[35]. There are four main steps:

- The pre-trained word should be embedded.
- An object tokenizer should be created.
- Text documents should be transformed into token sequences and then padded
- Then token mapping will be created and then embedded[18].

4.6 Data Splitting

The dataset is divided into two parts: train and test. Also, the dataset is labeled as 0 and 1, where 0 is for fake news, and 1 is for real news, and the section comprising 70% of the dataset is trained data. Here, the algorithm detects real news and false news. Whether the news is real or fake, return it if it is correct or incorrect, and build the algorithm's percentage based on the ratio of correct and incorrect answers.

4.7 Model Classification

Here, we have developed three classifiers for detecting fake news in Kurdish. Various classifiers are fed the extracted features. Additionally, classic classifiers like SVM and RF, as well as neural network methods like CNN from Sklearn, are used in our study. All of the classifiers utilized each of the extracted features. In this study, we evaluate the performance of three different classifiers based on the count-vectorizer, TF-IDF, and word embedding feature extraction techniques. Accuracy, recall, precision, and the F1 score are some examples of evaluation criteria. After the model was fitted, we compared the accuracy and f1 score. Two of the best-performing models were chosen as candidate models for detecting fake news after fitting all the classifiers. The following sections explain all the algorithms in more detail. As we have mentioned earlier, several machine learning algorithms have been selected to test the performance of the dataset and discover the outperforming algorithms. The following algorithms are used:

- Random Forest (RF)
- Support Vector Machine (SVM)
- Convolutional Neural Network (CNN)

4.7.1 Support Vector Machine (SVM)

Support Vector Machine (SVM) is an algorithm in supervised learning. It is the most popular algorithm for classification and regression problems. The data are transformed using a method known as the "kernel trick," Based on these transformations, it determines the most suitable output boundary. Because they choose the decision boundary that maximizes the distance from the nearest data points of all the classes, SVMs differ from other classification techniques[36]. The test classifier uses each feature extraction method to determine whether it produces the desired results. In addition, the default parameters of the SVM classifier were employed without parameter tuning or modifying the default parameters like:

C=1.0

Kernel=' rbf'

Gamma=' scale'

Degree=3

class_weight=None

max_iter=-1

random_state=None

4.7.2 Random Forest (RF)

Random forests, also known as random decision forests, are ensemble learning techniques for classification, regression, and other tasks that operate by constructing work by building a large number of decision trees during the training phase and then producing the class that represents the mean or average prediction or mode of the classification. In classification and regression issues, it is a supervised machine learning algorithm. On various samples, it constructs decision trees and uses their average for classification and a majority vote for regression[37]. To determine if the classifier yields the expected results, test it using each feature extraction technique. Additionally, there are many parameters in RF classifiers, but the RF classifier's default parameters were used without any parameter adjustment or modification, such as:

n_estimators=100

max_depth=None

random_state=None

verbose=0

class_weight=None

ccp_alpha=0.0

4.7.3 Convolutional Neural Networks (CNN)

Machine learning includes convolutional neural networks, also known as CNNs. It is a subset of the several artificial neural network models employed for various applications and data sets. A CNN is a particular type of network architecture for deep learning algorithms utilized for tasks like image recognition and pixel data processing[38]. However, it is still successful in text processing today. A convolutional neural network provides promising outcomes for the challenge of classifying texts. The only difference between text and image classification criteria is that "vector matrices" is used instead of "pixel values." Most NLP tasks use phrases or documents encoded as a matrix rather than image pixels as input. The matrix's rows correspond to a single token, which is usually a word but could also be a character[39]. In addition, the CNN model was the third one. The output of the initial embedding layer was fed into a one-dimensional convolution layer with 100 filters, a kernel size of 2, and the activation function ReLu. The GlobalMaxPooling pooling layer and a dense layer with 260 neurons and a ReLu activation function were added after this. A sigmoid activation function and one neuron were present in the output layer. We employed binary cross-entropy and Adam optimization with 5 epochs as the loss function.

4.8 Evaluation Metrics

Four well-known classification metrics have been used such as Precision, Recall, F1-score, and accuracy, used in the evaluation stated in the formulas (4, 5, 6, 7) as follows[40]:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \quad (4)$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \quad (5)$$

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (6)$$

$$\overset{Accuracy}{=} \frac{TruePositive + TrueNegative}{TruePositive + TrueNegative + FalsePositive + FalseNegative} \quad (7)$$

Where:

True Positive: the number of false news that was successfully identified as such.

True Negative: the number of true news identified as true.

False Positive: The anticipated class is wrong, but the actual class is true.

False Negative: The anticipated class is true, but the actual class is incorrect[41].

Another evaluation metric for binary classification problems is the Receiver Operator Characteristic (ROC) curve, presented as a plot. The performance of the classification model at each classification threshold is shown in this plot. In addition, two parameters shown on this curve are TPR and FPR[13].

The final measurement is the loss function. It is defined as the error between the output of our algorithms and the specified target value calculated using loss functions. In deep learning tasks, the loss function usually measures the accuracy, similarity, or goodness of fit between the predicted value and ground truth. A correctly constructed loss function can significantly improve the training performance of the neural network. In other words, loss functions can be divided into two main categories: classification and regression[42]. There are some types of loss function classification algorithms. In our study, we used binary cross-entropy loss in CNN architecture because of the problem of classifying us into two classes. Also, the value of the CNN model with % epochs is 0.241. In addition, the data used in this study, labeled into two categories, is "real" and "fake".

4.9 Experimental Setup

This section describes the experimental setups used to perform the text categorization task. We designed experiments to validate and guarantee the quality of manually and automatically generated annotations. We investigated fake news detection as a binary classification (fake or real). As a result, divide the KDFND dataset into training and testing, with 70% of the dataset for training and 30% for testing. Three machine learning and deep learning classifiers were used to classify the dataset as fake news: the Support Vector Machine (SVM), the Random Forest Model (RF), and the Convolutional Neural Network (CNN).

4.10 Models Training (MT)

When the numerical form of the textual news was completed, the data frame containing the count vector, TF-IDF, and word embedding representations for each news item in the KDFND dataset was used to train three distinct classifiers. The proposed algorithm is carried out utilizing Python 3.9. The functions offered in a Jupyter Notebook environment for various classifiers can be used with the help of the Sklearn, Numpy, pandas, mat-plot lib, seaborn, and os libraries.

5. Results and Discussion

In this section, the performance analysis of classifiers on the KDFND dataset has been described. The dataset was split into two parts randomly. One part contains 70% of the data, which is used to train the classifiers, and the remaining 30% is used to test the performance of the classifiers. Furthermore, the experiments were divided into three sections based on the type of feature extraction (TF-IDF, Count-Vector, Word-Embedding). The first experimental results on the KDFND dataset are presented. We used three machine learning and deep learning classifiers (SVM, RF, and CNN). Table 4 shows the performance of the three distinct models using the abovementioned metrics.

Additionally, the Support Vector Machine algorithm obtained a better (80.9% accuracy and 81.17% F1 score) for the test dataset with the TFIDF feature than the other classical algorithms when compared to other classical algorithms. The deep learning models performed similarly to each other. The CNN architecture outperformed the other models with an accuracy of 81.9% and an F1 score of 82.4% for the test dataset.

Table 4: Performance classifiers with TF-IDF.

ML\MT	TF-IDF			
	F1	Precision	Recall	Accuracy %
SVM	81.17	80.43	86.17	80.9
RF	80.83	82.55	80.85	80.83
CNN	82.4	82.67	80.22	81.9

The findings of the second experiment on the KDFND dataset are given. Using three machine learning and deep learning classifiers (SVM, RF, and CNN), Using the various measures discussed above, According to Table 5, the Random Forest approach outperformed the other traditional techniques for the test dataset in terms of accuracy (80.36%) and F1 score (81.17% from SVM). And in contrast to other traditional algorithms. There were only slight variations in performance amongst the deep learning models; with an F1 score of 80.46% percent for the test dataset and an accuracy of 80.55%, the CNN architecture surpassed the other models. Using the count-vector feature extraction approach to produce features

Table 5: Performance classifiers with Count-Vector.

ML\MT	Count Vector			
	F1	Precision	Recall	Accuracy %
SVM	80.09	80.43	79.94	79.96
RF	81.17	81.6	80.36	80.36
CNN	80.46	80.32	80.4	80.55

The third experiment's findings about the corpus of fake news in Kurdish are presented. We labeled the datasets manually and automatically using three machine learning and deep learning classifiers (SVM, RF, and CNN). Using the measures discussed above, Table 6 demonstrates that the Random Forest algorithm outperformed other traditional algorithms for the test dataset (73.37% accuracy and 74.17% F1 score) while using the word-embedding method. The deep learning models have differences in performance from the other models. The CNN architecture outperformed the other models with a 91.6% accuracy and an F1 score of 95.47% for the test dataset. Additionally, Figure 6 displays the classification performance metrics for each algorithm.

Table 6: Performance classifiers with word-embedding.

ML\MT	Word Embedding			
	F1	Precision	Recall	Accuracy %
SVM	73.6	71.38	82.93	73.1
RF	74.17	73.41	75.84	73.37
CNN	95.47	93.68	97.34	91.6

Finally, our experiments used TF-IDF, Count-Vector, and word embedding techniques with SVM, RF, and CNN classifiers. The relationship between the true positive rate and the false positive rate is well-represented by the ROC curve. The SVM had the best ROC in our experiments choosing the TF-IDF feature with classifiers to present the ROC curve (0.88).

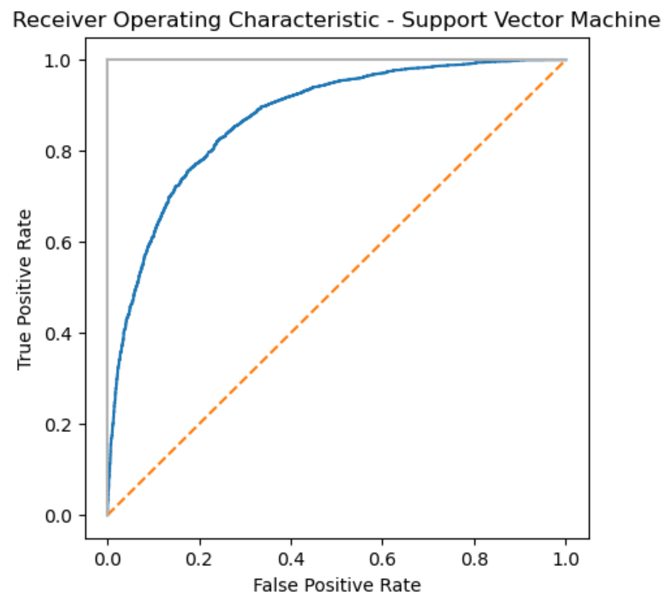

Figure 5: ROC curves of the SVM classifier.

The SVM classifier's learning curves are shown in Figure 4. The training curves converging at a high score demonstrates that the model does not suffer from overfitting or underfitting issues.

6. Benchmarking with previous studies

The dataset contains no additional research. It is the first research on the dataset because we compared the proposed system to another study that differs from the data set. The latest and most relevant paper that used the same feature extraction methods was chosen to compare with other studies. Additionally, the comparison between the proposed system and studies was made based on the accuracy results of different classification algorithms. We conducted a model benchmark with comparable studies in the Kurdish, Arabic, and Persian languages to increase our work's credibility and value. However, we used all three machine learning approaches with default values without optimization or hyperparameter tuning, the purpose of which is essential research for future research. We also selected metrics that can be used to compare machine learning classifiers across languages. Table 7 represents a comparison study with another related work.

Table 7: Benchmarking our model with metrics performance.

Algorithm /related works	Accuracy% S. S. Alanazi and M. B. Khan[14]	Accuracy% Zarharan M. et al. [19]	Accuracy% Azad R. et al. [11]	Accuracy% Jardaneh G. et al [3]	Accuracy% Thaher H. et al. [5]	F1% M. A. Bsoul. et al. [15]	Proposed model accuracy%
SVM	72.50	64	88.71	-	75.39	70-80	80.9
RF	-	68	86.34	76	75.98	60-70	81.87
CNN	-	-	-	-	-	-	91.7

Finally, as was mentioned earlier, only one[11] study in the literature applied several machine learning algorithms for fake news detection to a tiny dataset for Kurdish. The authors prepared two sets of news: the first set includes (5000) real and (5000) fake news from Facebook pages and Sorani Kurdish websites, and the second set contains (5000) the same real news and (5000) fake news that was scraped from real news and is considered fake news in this set. On the other hand, they didn't use any deep learning algorithms. This study was compared to earlier studies related to Kurdish languages. Although their dataset was very small and only considered social media platforms, their best CA with SVM results was only 88.71%. In contrast, this study's CA with CNN on the significantly larger dataset was 91.6%. It is proven that this study makes a significant contribution to the field of fake news detection by creating the largest online corpus and proposing the new, outperforming CNN algorithm.

Conclusion and Future Work

Online and social media forums have increased in popularity, and the dissemination of fake news has become a major concern for many businesses and organizations. It can be difficult to identify fake news on social media networks. Nevertheless, considering that Kurdish is a very complex language, a few studies have been published to address the Kurdish context. In this paper, fake news is classified using a framework created using text data from Kurdish news articles. In this study, we focused on and used the publicly available Mendeley Data Fake News dataset, which consists of news articles from multiple domains and contains 100,962 news articles, of which 50,751 are real, 50,211 are fake news labeled 1 and 0. We analyzed the dataset using two machine learning classifiers and one deep learning classifier, like Random Forest, Support Vector Machine, and Convolutional Neural Network. After applying pre-processing steps, we extracted different textual features from the articles using the TF-IDF, count-vector, and word-embedding feature extraction techniques. We used the feature set as fed into the classifiers. Using the abovementioned classifiers and features, we introduced a multi-model fake news detection system using various models to get more accurate findings. The research paper results show the performance of the CNN architecture was superior. It could identify the most incorrect articles while producing fewer false negatives and complete the classification task with a higher accuracy of 91% and a higher f1-score of 95%. Finally, this study establishes a baseline for future experiments and broadens the available options for detecting fake news. The dataset size should be increased or decreased to find more relationships and patterns, and more classifier experiments should be run.

Conflict of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author Contribution

Both authors of this paper participated equally in every stage of this study's drafting, reviewing, and submission.

Funding Information

We declare that This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Acknowledgments

The authors thank SPU for providing the opportunity and supporting this study.

References

1. X. Zhang, A. A. J. I. P. Ghorbani, and Management, "An overview of online fake news: Characterization, detection, and discussion," vol. 57, no. 2, p. 102025, 2020.
2. L. Shree, N. K, R. M, S. K, and T. R, "Fake News Detection Using Machine Learning," *International Research Journal of Computer Science,* vol. 9, no. 8, pp. 217-221, 2022.
3. G. Jardaneh, H. Abdelhaq, M. Buzz, and D. Johnson, "Classifying Arabic tweets based on credibility using content and user features," in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 2019, pp. 596-601: IEEE.
4. S. Senhadji and R. A. J. I. I. J. o. A. I. San Ahmed, "Fake news detection using naïve Bayes and long short term memory algorithms," vol. 11, no. 2, p. 746, 2022.
5. T. Thaher, M. Saheb, H. Turabieh, and H. J. S. Chantar, "Intelligent detection of false information in Arabic tweets utilizing hybrid harris hawks based feature selection and machine learning models," vol. 13, no. 4, p. 556, 2021.
6. A. R. Mahlous, A. J. I. J. o. A. C. S. Al-Laith, and Applications, "Fake news detection in Arabic tweets during the COVID-19 pandemic," vol. 12, no. 6, pp. 778-788, 2021.
7. V. Agarwal, H. P. Sultana, S. Malhotra, and A. J. P. C. S. Sarkar, "Analysis of classifiers for fake news detection," vol. 165, pp. 377-383, 2019.
8. R. Setiawan *et al.*, "Certain investigation of fake news detection from Facebook and Twitter using artificial intelligence approach," pp. 1-26, 2021.
9. R. O. Abdulrahman, H. Hassani, and S. J. a. p. a. Ahmadi, "Developing a fine-grained corpus for a less-resourced language: the case of Kurdish," 2019.
10. H. Hassani, D. J. C. S. Medjedovic, and I. Technology, "Automatic Kurdish dialects identification," vol. 6, no. 2, pp. 61-78, 2016.

11. R. Azad, B. Mohammed, R. Mahmud, L. Zrar, S. J. T. J. o. C. Sdiqa, and M. Education, "Fake News Detection in low-resourced languages "Kurdish language" using Machine Learning algorithms," vol. 12, no. 6, pp. 4219-4225, 2021.
12. M. Amjad, S. Butt, H. I. Amjad, A. Zhila, G. Sidorov, and A. J. a. p. a. Gelbukh, "Overview of the shared task on fake news detection in Urdu at Fire 2021," 2022.
13. M. ALAZAB *et al.*, "FAKE-NEWS DETECTION SYSTEM USING MACHINE-LEARNING ALGORITHMS FOR ARABIC-LANGUAGE CONTENT," vol. 100, no. 16, 2022.
14. S. S. Alanazi, M. B. J. I. J. o. C. S. Khan, and N. Security, "Arabic fake news detection in social media using readers' comments: Text mining techniques in action," vol. 20, no. 9, pp. 29-35, 2020.
15. M. A. Bsoul, A. Qusef, and S. J. P. C. S. Abu-Soud, "Building an optimal dataset for Arabic fake news detection," vol. 201, pp. 665-672, 2022.
16. H. Najadat, M. Tawalbeh, R. J. I. J. o. E. Awawdeh, and C. Engineering, "Fake news detection for Arabic headlines-articles news data using deep learning," vol. 12, no. 4, 2022.
17. S. Zamani, M. Asadpour, and D. Moazzami, "Rumor detection for Persian tweets," in *2017 Iranian Conference on electrical engineering (ICEE)*, 2017, pp. 1532-1536: IEEE.
18. M. M. J. T. J. o. C. Sadr and M. Education, "The use of LSTM neural network to detect fake news on Persian Twitter," vol. 12, no. 11, pp. 6658-6668, 2021.
19. M. Zarharan *et al.*, "Persian Stance Classification Data Set," in *TTO*, 2019.
20. M. H. Saghayan, S. F. Ebrahimi, and M. Bahrani, "Exploring the impact of machine translation on fake news detection: a case study on Persian tweets about COVID-19," in *2021 29th Iranian Conference on Electrical Engineering (ICEE)*, 2021, pp. 540-544: IEEE.
21. D. A. Salh and R. Nabi, "Kurdish Dataset for Fake News Detection(KDFND)," V1 ed. Mendeley Data, 2022.
22. R. Oshikawa, J. Qian, and W. Y. J. a. p. a. Wang, "A survey on natural language processing for fake news detection," 2018.
23. H. Himdi, G. Weir, F. Assiri, H. J. A. J. f. S. Al-Barhamtoshy, and Engineering, "Arabic fake news detection based on textual analysis," vol. 47, no. 8, pp. 10453-10469, 2022.
24. S. Ahmadi, "KLPT–Kurdish language processing toolkit," in *Proceedings of the second workshop for NLP open source software (NLP-OSS)*, 2020, pp. 72-84.
25. A. M. Mustafa and T. A. J. J. o. I. S. Rashid, "Kurdish stemmer pre-processing steps for improving information retrieval," vol. 44, no. 1, pp. 15-27, 2018.
26. J. Singh and V. J. A. C. S. Gupta, "Text stemming: Approaches, applications, and challenges," vol. 49, no. 3, pp. 1-46, 2016.
27. C. M. M. Kotteti, X. Dong, N. Li, and L. Qian, "Fake news detection enhancement with data imputation," in *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, 2018, pp. 187-192: IEEE.
28. T. Chauhan and H. J. I. J. o. I. M. D. I. Palivela, "Optimization and improvement of fake news detection using deep learning approaches for societal benefit," vol. 1, no. 2, p. 100051, 2021.
29. H. Reddy, N. Raj, M. Gala, A. J. I. J. O. A. Basava, and Computing, "Text-mining-based fake news detection using ensemble methods," vol. 17, no. 2, pp. 210-221, 2020.
30. S. Chaturvedi, A. Saxena, A. Gupta, and F. Alam, "Detecting Fake News using Machine Learning Algorithms," in *2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 2021, pp. 212-216: IEEE.
31. B. B. K. Bharathi C1, Anusha S T3, Aishwarya B N4, "FAKE NEWS DETECTION USING MACHINE
LEARNING," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 11, no. 6, 2022.
32. C. Zhang, A. Gupta, C. Kauten, A. V. Deokar, and X. J. E. J. o. O. R. Qin, "Detecting fake news for reducing misinformation risks using analytics approaches," vol. 279, no. 3, pp. 1036-1052, 2019.
33. M. J. Awan *et al.*, "Fake news data exploration and analytics," vol. 10, no. 19, p. 2326, 2021.
34. F. Gereme, W. Zhu, T. Ayall, and D. Alemu, "Combating Fake News in "Low-Resource" Languages: Amharic Fake News Detection Accompanied by Resource Crafting. Information 2021, 12, 20," ed: s Note: MDPI stays neutral concerning jurisdictional claims in …, 2021.
35. Y. Fang, J. Gao, C. Huang, H. Peng, and R. J. P. o. Wu, "Self multi-head attention-based convolutional neural networks for fake news detection," vol. 14, no. 9, p. e0222713, 2019.
36. M. G. Hussain, M. R. Hasan, M. Rahman, J. Protim, and S. Al Hasan, "Detection of Bangla fake news using MNB and SVM classifier," in *2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*, 2020, pp. 81-85: IEEE.
37. A. Abdulrahman and M. Baykara, "Fake news detection using machine learning and deep learning algorithms," in *2020 International Conference on Advanced Science and Engineering (ICOASE)*, 2020, pp. 18-23: IEEE.
38. K. Ivancová, M. Sarnovský, and V. Maslej-Krcšňáková, "Fake news detection in Slovak language using deep learning techniques," in *2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2021, pp. 000255-000260: IEEE.
39. D. Mwiti. (2021, Nov. 01, 2022).). *NLP Essential Guide: Convolutional Neural Network for Sentence Classification*.
40. M. Sarnovský, V. Maslej-Krešňáková, and N. Hrabovská, "Annotated dataset for the fake news classification in Slovak language," in *2020 18th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, 2020, pp. 574-579: IEEE.
41. C.-M. Lai, M.-H. Chen, E. Kristiani, V. K. Verma, and C.-T. J. A. S. Yang, "Fake News Classification Based on Content Level Features," vol. 12, no. 3, p. 1116, 2022.
42. C. Pere. (2020, 2023-03-28). *What is Loss Function*? Available: https://towardsdatascience.com/what-is-loss-function-1e2605aeb904