Social Network Analysis and Mining (2023) 13:16
https://doi.org/10.1007/s13278-022-01019-y

ORIGINAL ARTICLE

Early multi-class ensemble-based fake news detection using content features

Sajjad Rezaei[1] · Mohsen Kahani[1] · Behshid Behkamal[1] · Abdulrahman Jalayer[1]

Received: 30 October 2022 / Revised: 19 December 2022 / Accepted: 20 December 2022 / Published online: 31 December 2022

Abstract

Nowadays, social media plays an essential role in spreading the news with low cost and high speed in publishing, and easy availability. Given that, anyone can publish any news on social networks, with some of them to be fake. These fake stories should be detected as soon as possible since they might have negative impacts on the society. To address this issue, most researches consider fake news detection as a binary classification problem. However, as some news are half-true, recently, multi-class detection has gained more attention. This paper investigates an early detection of fake news using multi-class classification. This is achieved by extracting the content features, such as sentiment and semantic features, from the news. The proposed model employs five classifiers (Random Forest, Support Vector Machine, Decision Tree, LightGBM, and XGBoost) as primary classifiers. Furthermore, AdaBoost is used for the meta-learning algorithm to develop a stacking generalization model. Stacking generalization is an ensemble learning method that uses all data produced by the first-level algorithms. We trained our model with PolitiFact data for the evaluation, and the model performance was evaluated by Accuracy, Precision, Recall, and F1 score. Excremental evaluation of the real-world datasets showed that our proposed model outperformed all previous works in both binary and multi-class classifications.

Keywords Natural language processing · Machine learning · Fake news detection · Ensemble learning

1 Introduction

Today, social media has become an essential part of human life. Since the conventional news sources like newspapers and television are not interactive, their importance have been decreased. Hence, in this area, social networks such as Twitter and Facebook are the most popular. In these social networks, people can easily interact with together and publish their posts like personal information and, also, news. However, these news in social networks are not controlled in terms of trustworthiness; hence, they cannot be trusted. In some cases, this fake news may have a very destructive effect on a society. It is therefore critical to detect this news early before it spreads broadly.

In social media, people spread the news without awareness of its validation consciously or unconsciously. The story may be circulated thousands of times without verification due to a catchy title. For example, the fake news that Barack Obama was injured in an explosion caused the value of US stocks to fall by $ 130 billion (Rapoza 2017).

Abusers may spread false information in order to benefit financially or politically, such as publishing a story to reduce the popularity of an electoral rival or to increase the popularity of an electoral partner. Therefore, reliable detection methods should be developed to prevent these evil intents, which can help people not fall into these kinds of traps. One of the most prominent examples of how fake news affects the society is US presidential election in 2016. In the election, most US residents were exposed to fake articles about Trump more than Clinton. Therefore, these articles contributed to

Sajjad Rezaei, Mohsen Kahani, Behshid Behkamal and Rahman Jalayer authors contributed equally to this work.

✉ Mohsen Kahani
kahani@um.ac.ir

Sajjad Rezaei
sajjad.rezaei@mail.um.ac.ir

Behshid Behkamal
behkamal@um.ac.ir

Abdulrahman Jalayer
rahman.jalayer@mail.um.ac.ir

[1] Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad 9177948974, Khorasan Razavi, Iran

Trump's rise in popularity. Finally, the analysis of the poll result showed that Republican voters were generally more inclined to fake news articles (Allcott and Gentzkow 2017).

The main features of the news are shown in news content, and occasionally, at the beginning of its spread, just these features are present. Hence, extracting valuable content features aids to identify the fake news. All inventions and technologies are developing; as the same way, fake news follows this rule as well. Therefore, the fake news written style is continuously evolving, as are the methods of identifying the fake news. Hence, detecting the fake news becomes problematic issue, especially, when the style of fake news is similar to the real news.

Recently, we have seen that the labeling on news fact-checking websites has switched from binary to multi-class. Also, the information was previously gathered in datasets with labels that were either true or false. However, to give fake news greater credibility with readers nowadays, fake news writers mix true and false paragraphs together. This is a justification for offering algorithms that can classify fake news with multiple classes. The fake paragraph(s) is intentionally written for malicious purposes. Developing multi-class classification models is obviously essential for identifying fake news that is written as explained.

In previous works (Zhou, Jain et al. 2020), (Huang and Chen 2020), and (Agarwal and Dixit 2020), the news was only classified in binary form (the fake news datasets are multi-class in some cases, but sometimes specifying the threshold or removing other labels was changed to binary form and used to build the models). Although developing models with binary output can be beneficial, we cannot expect proper efficiency with changes in the structure of fake news. With the emergence of fake news that can be classified into something other than true and false, approaches have started to shift toward multi-class identification.

The results of classification algorithms such as SVM, Random Forest, and XGBoost, which were previously used to identify fake news by these papers (Zhou, Jain et al. 2020), (Huang and Chen 2020), and (Agarwal and Dixit 2020), show that each of these algorithms has a weakness in identifying, as demonstrated by model performance. As a result, ensemble learning can compensate for the shortcomings of the previous model by combining several weak algorithms to form a stronger network. Because the weaknesses of each algorithm are eliminated (via ensemble learning), the resulting model can be highly efficient and identify multi-class fake news with fewer errors than each single-algorithm models.

We used news collected from three reputable fact-checking websites and built a multi-class model to identify fake news. Therefore, we do not miss any information from the news that has labels different from true and false.

The main contributions of this paper are summarized as follows:

- Using the most useful linguistic features like textual, sentiment, semantic, and readability features
- Employing a new dataset that contains multi-source and multi-class fake news on a variety of topics
- The proposed model uses a Stacking ensemble network with five main classifiers (Random Forest, SVM, Decision Tree, LGBM, and XGBoost)
- Experiments were performed in both binary and multi-class, then compared with each related literature.

The remainder of this paper is organized as follows: In Sect. 2, the related literature is reviewed. Then, the classifier algorithms are briefly introduced in Sect. 3. In Sect. 4, the ensemble learning methods are presented. How the model is developed is elaborated in Sect. 5. Finally, the experimental results are reported, and the conclusion is drawn in Sects. 6, respectively.

2 Related works

Linguistic features are extracted from news content text directly and find essential information about fake news. There are many valuable Natural Language Processing (NLP) tools to analyze the collected information that can help us to fake news detection. For instance, Stanford CoreNLP[1] and NLTK[2] are NLP tools to extract knowledge from text. Vicario et al. (Vicario, Quattrociocchi et al. 2019) worked on polarization and bias between social media users who spread fake news online. In this paper, they tried to early polarize content detection and extracted textual features like semantic features and sentiment features. The number of characters, words, and sentences belongs to semantic features. They used linear regression, logistic regression, support vector machine, and K-nearest neighbor for classification.

Zhou et al. (Zhou, Jain et al. 2020) investigated a theory-driven model of fake news detection, which proposed four different levels for content features. Characters per word, Sentences per paragraph, word per sentence, and Number of sentences used as Quantity features. Diversity features had considered, such as the percentage of unique words and verbs, and sentiment features like the percentage of positive comments. These are some content features that important for fake news detection. They provided a complete list of semantic features involved in their study.

[1] https://stanfordnlp.github.io/CoreNLP/.

[2] https://www.nltk.org.

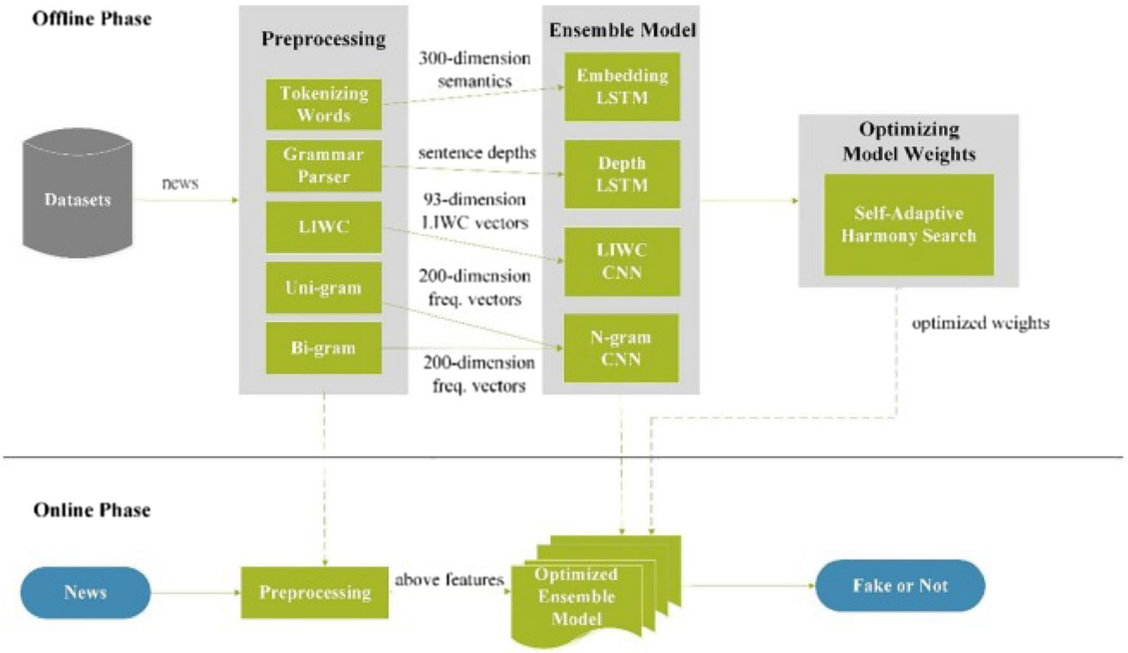

Fig. 1 Huang et al. (Huang and Chen 2020) proposed a model schema

Their experiments have been conducted on PolitiFact and BuzzFeed datasets with binary labels. Classifiers used in their paper are SVM, RF, and XGBoost.

Message-based features are one of the features that were considered to detect fake news by Castillo et al. (Castillo, Mendoza et al. 2011). It has been pointed out that message-based features can be Twitter-dependent or independent of Twitter. Twitter-independent features such as message length, question mark, and the number of emotional words are positive or negative. The Twitter-related quality mentions the hashtag and retweets the message. Unique sentiment features used in this paper are positive and negative word count in terms of sentiment and total sentiment score.

Rashkin et al. (2017) compared the language of real news with humor, hoaxing, and advertising to obtain unreliable text features. They reached the language of real news with humor and hoaxing to obtain unreliable text features. In this study, PolitiFact data with six labels were used. PolitiFact statements often have a degree of honesty between true and false. It has been pointed out that false information follows a slight difference rather than an obvious structure. They used NLTK tools for lexicon analysis and reported that words exaggerate and superlative adjectives and modal verbs are used more in writing fake news. Content features have valuable and significant information to identify fake news written style. Finally, they reported macro averaged F1 score for development set on PolitiFact 2-class and 6-class data. The best performance was LSTM without LIWC features for 2-class data and Maximum Entropy (MaxEnt) for 6-class data.

Huang et al. (Huang and Chen 2020) used a deep learning model to identify fake news. The proposed model is ensemble learning networks with four training models. Models are N-gram CNN, LIWC CNN, depth LSTM, and embedding LSTM and optimizing ensemble learning weights with Self-Adaptive Harmony Search (SAHS) techniques, as shown in Fig. 1. This SAHS algorithm is used to reach higher accuracy in the fake news detection model. After preprocessing step, they extract Uni-gram and Bi-gram for N-gram CNN. Besides, LIWC was used to get eigenvector of news for LIWC CNN, Grammar Parser produced sentence depths for depth LSTM, and embedding LSTM used tokenizing words. Three datasets (BuzzFeed, Satire, and PolitiFact) are used in this work.

Agarwal et al. (2020) implemented an ensemble learning approach to merge various classification models with SVM, Naïve Bayes, k-nearest neighbor, CNN, and LSTM. They said combing these models makes fake news detection more accurate. NB, KNN, and LSTM classifiers were trained on two datasets (Liar and collected dataset from Kaggle) and tried to produce credibility scores in short news. The Lair dataset has six different labels, and this paper sets a threshold to change labels to binary form and then uses the binary dataset to make its model.

When investigating the related works, it turns out that they did not use up-to-date datasets. Almost all classifications were formed in binary, which can affect the efficiency of the built models. Fake news written form is progressing over time. Therefore, new models are needed to identify fake news outside of the binary format.

3 Classifiers

In this section, the basis of the algorithms used in this paper is described briefly.

3.1 Random forest

A Random Forest (RF) algorithm fits several classifier trees on each dataset sample. For classification problems, the final class is selected by most trees. The number of trees built in the forest is controlled by "n-estimators" parameters (Ho 1995).

3.2 SVM

Support Vector Machine (SVM) is a supervised machine learning algorithm that can be used for classification. The SVM makes one or set of hyper-planes in N-dimensional, and the planes classify the training data points of any class. The formula present in the following poses the optimization issue solved by SVM (Bishop and Nasrabadi 2006).

$$\min_{\omega,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=0}^{n} \zeta \tag{1}$$

$$\gamma_i \left(w^T \phi \left(x_i \right) + b \right) \geq 1 - \zeta_i \tag{2}$$

where $\mathrm{w}^{\mathrm{T}}\mathrm{w}$ represents the average vector, C is a regularization parameter, ζ_i represents the distance to the correct margin, $\phi\left(x_i\right)$ represents the transformed input space vector, b is a bias parameter, and γ_i represents the i-th target.

The most popular kernel functions are the linear, polynomial, Radial Basis Function (RBF), and sigmoid functions (Bishop and Nasrabadi 2006). We use SVM algorithms with RBF kernel that function is as follows:

$$k\left(x_i, x_j\right) = \exp\left(-\gamma \left\| x_i - x_j^2 \right\| \right) \tag{3}$$

3.3 Decision tree

The Decision Tree (DT) makes classification models in the form of a tree structure. DT is a supervised learning method that predicts the target class by learning rules derived from the data features. One of the essential parameters in the DT algorithm is called "criterion." These parameters used to calculate the quality of a split and supported criteria are "Entropy" and "Gini" (L. Breiman 1984). The following equation where give the Entropy p_i is the probability that a point is in the subset of the dataset:

$$\mathrm{Entropy}(P) = -\sum_{i=1}^{n} p_i \log_2 \left(p_i \right) \tag{4}$$

And Gini index is given by the below equation:

$$\mathrm{Gini}(P) = 1 - \sum_{i=1}^{n} (p_i)^2 \tag{5}$$

3.4 LightGBM

LightGBM is a gradient boosting algorithm that uses trees for classification. When there are many features in data, they can be used LGBM. It has several advantages like:

- Handling large dataset
- Efficient memory usage
- Faster training speed
- Higher accuracy

To solve the problem of a large number of features, uses Gradient-Based One Side Sampling (GOSS) or Exclusive Features bundling (EFB) (Ke, Meng et al. 2017).

3.5 XGBoost

XGBoost is one of the valuable and efficient distributed gradients boosting models. It is an end-to-end tree-boosting and supervised model. XGBoost prepares a parallel tree boosting that solves many classification problems accurately and fast (Chen and Guestrin 2016).

4 Ensemble learning

Ensemble learning is a machine learning technique that combines outputs of basic learner algorithms and produces one optimal prediction model (Opitz and Maclin 1999). Ensemble methods can be used to increase the outperformance of predictive models. The three typical ensemble learning methods are Stacking, Bagging, and Boosting.

Stacking comprises several different training models on the same data and uses another model to achieve the best combination of the prediction (Rokach 2010), as shown in Fig. 2 (a).

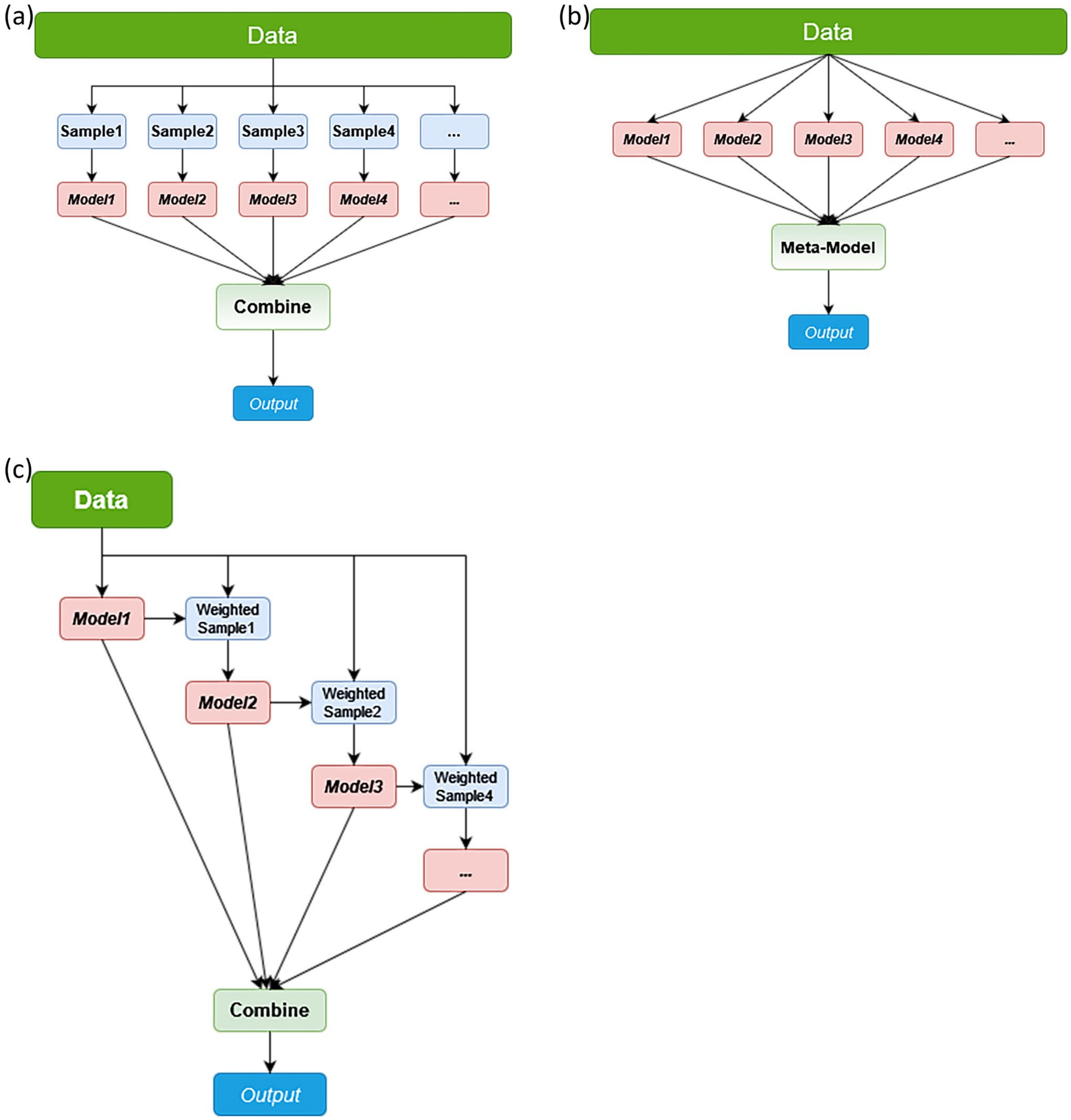

Fig. 2 shows different ensemble learning methods: **a** Stacking, **b** Bagging, and **c** Boosting

Bagging perhaps is the simplest method in ensemble learning with good efficiency. It uses different dataset samples and then combines them by taking a simple majority vote of the prediction (Rokach 2010), as shown in Fig. 2 (b).

Boosting trains models sequentially, where those methods mentioned above are trained in parallel. A new model is created and solving its incompetence in the previous step (Rokach 2010), as shown in Fig. 2 (c).

Table 1 Complete features list is extracted from the news

Features name	Category	Features name	Category
Similarities between the title and the text of the news	Semantic	Ann semantic score	Sentiment
Similarities between the topic modeling title and the text of the news	Semantic	Vader semantic score	Sentiment
Number of characters	Quantity	Positive words count	Sentiment
Number of words	Quantity	Negative words count	Sentiment
Number of sentences	Quantity	Coleman–Liau Index (CLI)	Readability
Number of capital words	Quantity	Gunning fog index (GFI)	Readability
Number of punctuations	Quantity	Automated Readability Index (ARI)	Readability
Avg. number of characters per word	Quantity	Flesch–Kincaid Grade Level (FKGL)	Readability
Avg. number of words per sentence	Quantity	Flesch Reading Ease Index (FREI)	Readability

5 Proposed approach

We create our model for early fake news detection in 3 main steps. Firstly, all the news in the dataset is preprocessed to make data ready for machine learning algorithms. Secondly, features are extracted from the title and the text of the news, for example, generated the new title for each news, and calculated similarities, sentiment scores, and quantity features are added to the list of content features that help identify fake news. Finally, a stacking ensemble network is built with five basic classifiers. Table 1 shows all components used in our model.

5.1 Data preprocessing

The raw data in the dataset are prepared for machine learning algorithms by removing worthless pieces. At the first step, all news (text and title of news) convert to lower case, then extra spaces are replaced with one space, and some special characters like Â €, ™ are removed. At last, all English stop words were released by the NLTK library.

A typical algorithm used to fit textual content into machine algorithms for prediction is TF-IDF. The news was converted to a numeric vector with TF-IDF and N-grams in our work. Vectors are made with 1-g, 2-g, 3-g, and 4-g.

5.2 Feature generation

Content-based features are one of the first and most essential features that can be used to detect fake news. So, the features that can produce from the news text include 1- Generate new title for the news (Topic modeling), 2- Calculate similarity, 3- Calculate sentiment score, 4- Quantity textual features.

5.2.1 Topic modeling

The primary purpose of the news title is to absorb the reader's attention. When people see the attractive title encouraged to read the whole news article, and also news title is used to affect the reader's discernment of fake news (Zhou and Zafarani 2018). To solve the problem of mismatching the news title with the news text to mislead the reader, we use Topic modeling with NLTK and Gensim libraries. We can find discussed topics in the news text and then compare them with the original news titles with topic modeling. We use WordNet of NLTK to understand the word's meaning and lemmatize the terms to get the root. Using Latent Dirichlet Allocation (LDA), find ten topics in the news, then find similarities between the original title and issues generated in the calculating similarity section with the topic modeling method (Li 2018).

It can be said that when the similarity between the original title of the news and the topics extracted from the text of the news is low, this news is prone to be fake because the author tried to create a wrong attitude in the reader by changing the title of the news. This is a novel feature that extracts from news content.

5.2.2 5.2.2 Calculate similarity

This part divides into two steps. Computing similarity between the original title and news text at the first step and the second step is similarity calculation between topic modeling output with the original news title. These two similarity values are added to the extracted features list.

5.2.3 5.2.3 Calculate sentiment score

We use two popular approaches for sentiment score calculation. Afinn (Nielsen 2011) is a word list approach for analyzing the sentiment of the text. The similarity score range produced by Afinn is from -526 to 282. Valence Aware Dictionary and sEntiment Reasoner (VADER) (Hutto and Gilbert 2014) is a rule-based tool for sentiment scores. This tool presented a negative sentiment score, a positive sentiment score, a neutral sentiment score, and a combined score

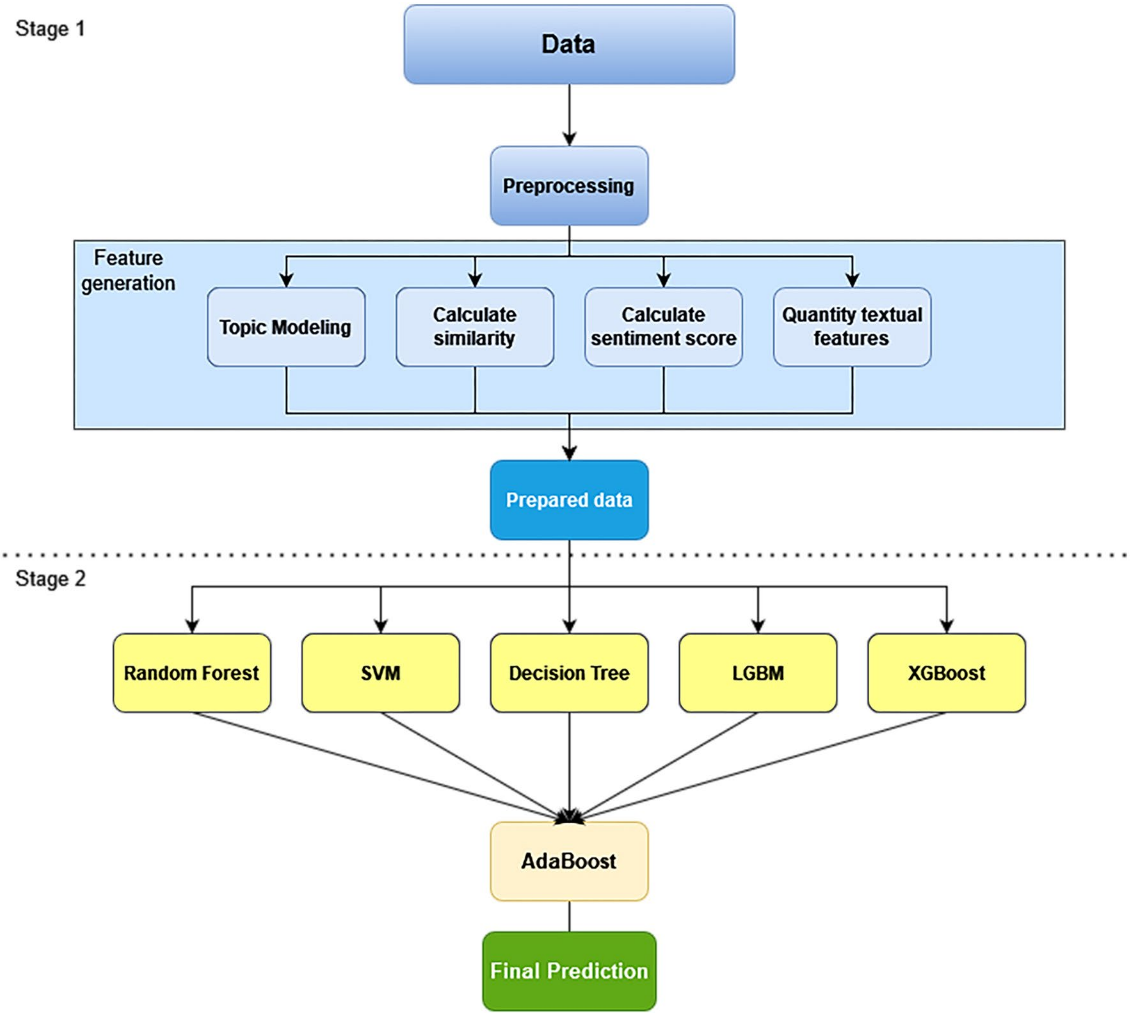

Fig. 3 Proposed model

for each news. The combined score was calculated by adding the three scores together and then normalizing between + 1 and -1. Also, the number of positive and negative words (from the perspective of sentiment) is counted and added to the list of features.

5.2.4 Quantity textual features

Fake news written style can be captured by collecting quantity features. These features include the number of characters, the number of words, and the number of sentences. If we want to show how much the news is complex can use the average of characters per word and words per sentence. The best features in the news text can be extracted when the goal is early detection. Therefore, we prepared a complete list of these features shown in Table 1 and used them to perform better represent fake news to machine learning algorithms (Zhou, Jain et al. 2020).

5.3 Building model

After preprocessing, vectors as output are prepared for classification, but there is another step. In feature extraction, we collected meaningful features which must be added to the prepared vectors. After doing this, those are ready to be used for machine training. Figure 3 has two stages; stage 1 shows preprocessing and features generation steps. Stage 2 represents an ensemble-based network with five basic classifiers (Random Forest, SVM, Decision Tree, LGBM, and XGBoost) that uses AdaBoost as a meta classifier. Figure 3 shows how to build a model from start to end.

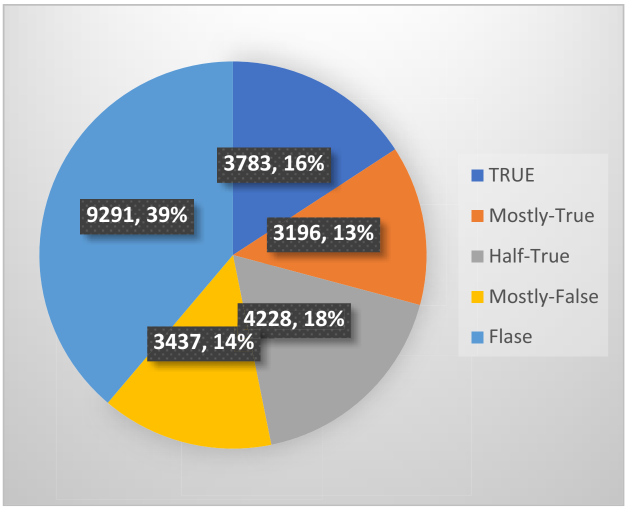

Fig. 4 statistic of the news in the dataset

Table 2 Cost per example

	True	Mostly true	Half-true	Mostly False	False
True	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	1
Mostly true	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$
Half-true	$\frac{1}{2}$	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{2}$
Mostly false	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	0	$\frac{1}{4}$
False	1	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	0

6 Experiments

6.1 Datasets

One of the most frequently used datasets in previous works is the Liar dataset (Wang 2017). Liar is a multi-class dataset with six different labels collected from traditional news outlets like TV or radio and campaigns. There are 12,836 short statements with 'True,' 'Mostly True,' 'Half True,' 'Barely True,' 'False,' and 'Pants of Fire' labels. The other valuable columns include statement, subjects, speaker, and speaker's job.

There are several fake news datasets like BuzzFeed corpus (Potthast et al. 2017), Satire dataset (Rubin, Conroy et al. 2016), CREDBANK (Mitra and Gilbert 2015), and FEVER (Thorne, Vlachos et al. 2018) that each of which has its characteristics, and researchers select all or part of them to use in their research. Still, one of the most striking features is that the news label is in binary format like True or False. For this reason, lots of papers built a binary fake news detection model. Multi-class datasets have been converted to binary datasets by thresholds and then used.

One of the contributions in our paper is a new dataset which was recently introduced that collected 24,517 news from three fact-checking websites (Politifact.com, Snopes.com, and TruthOrFiction.com) from September 1995 to January 2021. Most of the collected news is related to PolitiFact with about 60 percent of the dataset, then Snopes with 35 percent standing in second place, and TruthOrFiction with just about 5 percent is in last place. These websites have different methods to set the label for the news. There are five standard labels in this dataset (True, Mostly-True, Half-True, Mostly-False, and False) for each news. This dataset distinguishes itself from others with two features: multi-class and up-to-date news (Rezaei, Kahani et al. 2021).

We selected this dataset because of collecting new data from reliable websites. The fake news writing style is progressing, and this development impacts the detection, so the more recent data can help make an efficient detection model. The dataset consists of 23,935 (each row with a null value is removed), as shown in Fig. 4, and has five labels (True, Mostly-true, Half-true, Mostly-False, and False).

Our paper performs the experiments on Intel I7-4710HQ CPU, NVIDIA GeForce GTX 850 M GPU, and 12 GB memory. One of the famous python libraries for classification, regression, and clustering is Scikit-learn which we use to create first-level classifiers. First-level classifiers that we use to create an ensemble network are Random Forest, SVM, Decision Tree, LGBM, and XGBoost. The Scikit-learn library also creates the stacking ensemble network. The stacking network comprises the output of individual classifiers and uses a meta classifier to compute the final prediction.

6.2 Evaluation

Different metrics were provided for evaluation, such as Accuracy, Precision, Recall, and F1-Score, so we compare models with them.

Accuracy: Defined as the number of correctly predicted data instances over the total number of cases.

$$\text{Accuracy} = \frac{|TP| + |TN|}{|TP| + |TN| + |FP| + |FN|} \tag{6}$$

Precision: Defined as the proportion of correctly predicted positive instances to the total positive ones.

$$\text{Precision} = \frac{|TP|}{|TP| + |Fp|} \tag{7}$$

Recall: Defined as the proportion of correctly predicted positive instances to all instances in the actual class.

$$\text{Recall} = \frac{|TP|}{|TP| + |FN|} \tag{8}$$

Table 3 Evaluation metrics of each model

Model Metrics	SVM (%)	Decision Tree (%)	Random Forest (%)	XGBoost	LGBM
Precision	75	80	83	85%	85%
Recall	75	80	80	84%	84%
F1-Score	74	80	80	84%	84%
Accuracy	72	78	79	82%	84%

F1 Score: The weighted average of Precision and Recall.

$$F1 - \text{Score} = 2*\frac{\text{precision}*\text{recall}}{\text{precision} + \text{recall}} \tag{9}$$

Besides the most popular evaluation metrics mentioned above, Cost Per Example (CPE) is one of the metrics used to show how much multi-class models cost for a wrong prediction. A cost matrix is constructed to evaluate a multi-class fake news classification model, as shown in Table 2. In the multi-class classification, this matrix is better displayed. Equation (5) shows how to calculate the CPE (Toosi and Kahani 2007).

$$\text{CPE} = \frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{m} CM(i,j) * C(i,j) \tag{10}$$

where C(i,j) and CM(i,j) are confusion and cost matrices, N denotes the total number of test samples, and m represents the number of classes in the classification operation. A confusion matrix is a square matrix in which each column is assigned to an actual class, and each row is assigned to a predicted class. Each element in row i and column j, CM(i, j), indicated the number of not correctly classified samples that belonged to class i and were ranked in class j. The elements on the primary diameter of the matrix indicate the number of samples that are correctly classified. The cost matrix is structurally similar to the confusion matrix. Its values are set between zero and one, except that the C(i, j) element in this matrix is the penalty cost for incorrectly classifying an instance. Therefore, the principal diameters of matrix C always have a zero value because the principal diameter indicates the correct classification of the samples. The cost matrix is designed innovatively between 0 and 1, costing zero for the best case and costing one for the worst-case scenario. This matrix may not be the best, but it is a benchmark for evaluating our model. When the calculated penalty is closer to zero, the model receives a minor penalty and performs better (Toosi and Kahani 2007).

As shown in Table 3, the evaluation metrics (Precision, Recall, F1-Score, and Accuracy) of five basic classifiers are reported separately. While we describe our model with the Accuracy, Precision, Recall, and F1-score, some works only reported part of these evaluation metrics, so a comparison has been made with the reported metrics. Additionally, we employ fivefold cross-validation to train our model.

Table 4 Comparison of the result of our model on PolitiFact data (binary classification)

	PolitiFact			
	Accuracy	Precision	Recall	F1-score
(Huang and Chen 2020)	76	75	75	75
(Shu, Wang et al. 2019)	87	86	89	88
(Zhou, Jain et al. 2020)	89	87	90	89
(Palani, Elango et al. 2021)	93	92	91	92
Stacking Ensemble Network	**96.24**	**96.67**	**96.74**	**96.71**

Table 5 Comparison of the result of our model on PolitiFact data (multi-class classification)

Methods	PolitiFact			
	accuracy	Precision	Recall	F1-score
(Rashkin, Choi et al. 2017)	--	–	–	22
Stacking ensemble network	94.40	94.31	94.02	94.15

Table 6 Comparison of the result of our model

	All Data (PolitiFact + snopes + TruthOrFiction)			
	Accuracy	Precision	Recall	F1-score
The proposed model multi-class form	83.60	83.97	81.94	82.81
The proposed model binary form	91.52	93.21	93.65	93.43

We train our model with a multi-class dataset in two different ways. For evaluation with binary models built in (Huang and Chen 2020), (Shu, Wang et al. 2019), (Zhou, Jain et al. 2020), and (Palani, Elango et al. 2021) need to change the multi-class dataset to a binary dataset. Firstly, we convert labels of the dataset to binary form, which changes the mostly true label to true and mostly false to false and leave out the half-true labels then train the model with them. Evaluation report is shown in Table 4. Secondly, we train the model with multi-class data and compare it with (Rashkin, Choi et al. 2017), who provided only F1-Score for comparison. Still, all evaluation metrics of our model are reported in Table 5. Each previous work performed its model with different datasets, but PolitiFact data are common within all of them. Therefore, in one part of our experiments, we train our model on PolitiFact data and then evaluate it. Train and test sizes are 80% and 20%, respectively.

Table 7 Confusion matrix result of our model on multi-class data

		True	Mostly true	Half-true	Mostly false	False	CPE
Actual label	True	1641	112	38	53	4	0.0487
	Mostly True	112	600	41	23	3	
	Half-True	91	68	660	18	3	
	Mostly False	71	51	28	522	1	
	False	15	34	24	6	568	
	Prediction label						

Our primary intent is to build a multi-class fake news detection model with diverse PolitiFact, Snopes, and TruthOrFiction. These are the three most famous fact-checking websites (Rezaei, Kahani et al. 2021). After all experimental evaluations are reported, we describe the classification report on complete data in a multi-class form, as shown in Table 6. Also, the confusion matrix of this trained model brings in Table 7.

7 Conclusion

This paper introduces a fake news model that can predict the news with two labels (true and false) and five labels (true, mostly true, half-true, mostly false, and false). This model helps us to understand how much news is reliable. Combining the best text classification algorithms with ensemble learning networks is valuable because it can achieve a more accurate model than each model alone. Today, one of the difficulties in detecting fake news is that their writing style is more similar to the real news. The evaluation results show that the ensemble learning outputs of multi-class prediction on all training data and PolitiFact data are 83% and 94% as F1-score, respectively. Also, our binary form outperforms 96% as precision. The primary threat of our work is the lack of context features that can be used in combination with components extracted from news content to train the model.

Acknowledgements Not Applicable.

Authors' contributions The first author was involved in designing the initial idea, designing the implementation method and building the model and programming parts of the classification and ensemble learning algorithms. The second and third authors have collaborated in ideation, correction of the article and supervision of the work process. The fourth author was involved in correcting model programming errors and providing optimal solutions.

Funding 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.

Availability of data Everyone can see our supporting data in this link: https://github.com/Sajjad75/FakeNewsDetection-Ensemble-Approch.

Declarations

Competing interests The authors declare no competing interests.

Ethical approval and consent to participate This research received no specific funding from funding organizations in the public, commercial or non-profit sectors. Also, all the authors have participated in this research work with full satisfaction and all its parts have been reported with complete accuracy and transparency.

Consent for publication This research work can be published with the consent of all its authors.

Human and animal ethics Not Applicable.

References

Agarwal A and A Dixit (2020) Fake news detection: an ensemble learning approach. 2020 4th international conference on intelligent computing and control systems (ICICCS), IEEE

Allcott H, Gentzkow M (2017) Social media and fake news in the 2016 election. J Econom Perspect 31(2):211–236

Bishop CM, Nasrabadi NM (2006) Pattern recognition and machine learning. Springer

L Breiman, J F, R Olshen and C Stone (1984) "Classification and regression trees." from https://scikit-learn.org/stable/modules/tree.html#mathematical-formulation

Castillo C, et al. (2011) Information credibility on twitter. Proceedings of the 20th international conference on World wide web

Chen T and C Guestrin (2016) Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining

Ho T K (1995) Random decision forests. Proceedings of 3rd international conference on document analysis and recognition, IEEE

Huang Y-F, Chen P-H (2020) Fake news detection using an ensemble learning model based on self-adaptive harmony search algorithms. Expert Syst Appl 159:113584

Hutto C and E Gilbert (2014) Vader: a parsimonious rule-based model for sentiment analysis of social media text. Proceedings of the international AAAI conference on web and social media

Ke G, et al (2017) "Lightgbm: A highly efficient gradient boosting decision tree." Advances in neural information processing systems 30.

Li S (2018) "Topic modelling in python with NLTK and Gensim." from https://towardsdatascience.com/topic-modelling-in-python-with-nltk-and-gensim-4ef03213cd21

Mitra T and E Gilbert (2015) Credbank: a large-scale social media corpus with associated credibility annotations. Ninth international AAAI conference on web and social media

Nielsen F Å (2011) "A new ANEW: evaluation of a word list for sentiment analysis in microblogs." arXiv preprint arXiv:1103.2903

Opitz D, Maclin R (1999) Popular ensemble methods: an empirical study. J Artific Intell Res 11:169–198

Palani B et al (2021) CB-Fake: a multimodal deep learning framework for automatic fake news detection using capsule neural network and BERT. Multimedia Tools Appl 81(4):1–34

Potthast M, Kiesel J, Reinartz K, Bevendorff J, Stein B (2017) A stylometric inquiry into hyperpartisan and fake news. arXiv preprint arXiv:1702.05638.

Rapoza K (2017) "Can 'fake news' impact the stock market?" Forbes News.

Rashkin H et al (2017) Truth of varying shades: analyzing language in fake news and political fact-checking. Proceedings of the 2017 conference on empirical methods in natural language processing

Rezaei S et al (2021) The process of multi-class fake news dataset generation. 2021 11th international conference on computer engineering and knowledge (ICCKE), IEEE

Rokach L (2010) Ensemble-based classifiers. Artif Intell Rev 33(1):1–39

Rubin V L, et al (2016) Fake news or truth? Using satirical cues to detect potentially misleading news. Proceedings of the second workshop on computational approaches to deception detection

Shu K et al (2019) Beyond news contents: the role of social context for fake news detection. Proceedings of the twelfth ACM international conference on web search and data mining

Thorne J et al (2018) "Fever: a large-scale dataset for fact extraction and verification." arXiv preprint arXiv:1803.05355

Toosi AN, Kahani M (2007) A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers. Comput Commun 30(10):2201–2212

Vicario MD et al (2019) Polarization and fake news: early warning of potential misinformation targets. ACM Trans Web (TWEB) 13(2):1–22

Wang W Y (2017) "" Liar, liar pants on fire": a new benchmark dataset for fake news detection." arXiv preprint arXiv:1705.00648

Zhou X et al (2020) Fake news early detection: a theory-driven model. Digital Threats: Res Practice 1(2):1–25

Zhou X and R Zafarani (2018) "Fake news: a survey of research, detection methods, and opportunities." arXiv preprint arXiv:1812.00315 2

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.