DOI 10.15622/ia.22.4.3

M. GIDWANI, A. RAO

COMPARATIVE ANALYSIS OF RUMOUR DETECTION ON SOCIAL MEDIA USING DIFFERENT CLASSIFIERS

Gidwani M., Rao A. **Comparative Analysis of Rumour Detection on Social Media Using Different Classifiers.**

Abstract. As the number of users on social media rise, information creation and circulation increase day after day on a massive basis. People can share their ideas and opinions on these platforms. A social media microblogging site such as Facebook or Twitter is the favoured medium for debating any important event, and information is shared immediately. It causes rumours to spread quickly and circulates inaccurate information, making people uneasy. Thus, it is essential to evaluate and confirm the level of veracity of such information. Because of the complexities of the text, automated detection of rumours in their early phases is challenging. This research employs various NLP techniques to extract information from tweets and then applies various machine learning models to determine whether the information is a rumour. The classification is performed using three classifiers such as SVC (Support Vector Classifier), Gradient Boosting, and Naive Bayes classifiers for five different events from the PHEME dataset. Some drawbacks include limited handling of imbalanced data, difficulty capturing complex linguistic patterns, lack of interpretability, difficulty handling large feature spaces, and insensitivity to word order and context by using the above classifiers. The stacking approach is used to overcome the above drawbacks in which the output of combined classifiers is an ensemble with LSTM. The performance of the models has been analyzed. The experimental findings reveal that the ensemble model obtained efficient outcomes compared to other classifiers, with an accuracy of 93.59%.

Keywords: Rumor detection, Machine learning, Social Media, SVC, Gradient boosting, Naive Bayes.

1. Introduction. Social networks are undisputedly the most prevalent information access environment today. It is due to massive amounts of data and massive social network connections. While researching social networks, researchers attempt to answer some issues that emerge from the rich relationships that they contain [1]. The plethora of posting methods accessible on internet communities like Twitter, Weibo, Instagram, and Facebook facilitates message dissemination. It, however, adds to the fast distribution of unverified whispers and conspiracy theories, which frequently evoke accelerated, massive, but naïve social responses. Even so, the World Economic Forum's 2013 report has identified massive digital misinformation as a significant technical and global political risk. False reports had an impact on our economies, which are not impervious to the dissemination of deceit. There are two types of rumours: new rumours and long-standing rumours. New rumours arise during breaking news, as do "long-standing rumours" that have been debated for a long time [2]. Rumours will spread and increase visibility more than previously anticipated. According to statistics, there were nearly 320 million monthly active users on Twitter in November 2015, with 500 million tweets shared per day on average [3]. Social media platforms have grown in popularity, with users using them to share news and events and voice their opinions. The network's ease also fosters the spread of lies; For example, the fast dissemination of disinformation in a brief period can have negative things. However, determining the validity of rumours purely based on their textual substance can be challenging. As a result, people have begun to pay attention to the views expressed in relative responses. Knowing the stances that users adopt in reaction to specific rumours provides helpful insights, whereas some denying or questioning comments expose false rumours. [4, 5, 6].

Figure 1 depicts an example of a rumour that circulated during the recent COVID-2019 prompted India's Lockdown, alleging a decrease in pension payments to Indian citizens. The tale was unsubstantiated and wrong, and the Ministry of Finance of the Government of India needed to dispute it and provide clarification.

Fig. 1. Sample rumour and fact-checked in recent times

Therefore, it is crucial to be able to tell whether information elements posted on social media are real and legitimate, especially regarding current events. In this situation, building automated credibility detection systems is essential because the information seeker cannot tell true information from fake information. Nonetheless, there are several difficulties in determining the news' reliability. Nevertheless, there are several difficulties in determining the trustworthiness of news; these difficulties are caused by the limited datasets that include the aspects of the tweet that must be examined. Without technology, it might be challenging to identify accurate and reasonable postings or information. It necessitates time and work. Several researchers [7] use Twitter API to determine the creditability of the information. The limited size of tweets and casual wording is another difficulty. The phase of feature extraction and selecting the ideal function for constructing and executing the classification devices is the most difficult problem in the rumour detection task when using noisy data. Rumour identification aims to determine whether or not an inbound social media post contains rumours. It is simply a text classification problem.

Feature engineering is a critical component of text categorization because it is required to convert data into a machine learning-friendly framework. The primary problem with text-based rumour identification is a lower discovery rate, which occurs when algorithms incorrectly categorize fake material. It has a bad result on the system's detection rate & precision. In recent years, AI approaches have produced cutting-edge outcomes for various natural language processing tasks. The PHEME dataset, which includes rumour and non-rumour tweets on five incidents – Charlie Hebdo, the German plane tragedy, the Ottawa massacre, the Sydney siege, and Ferguson – is used in this investigation.

Author contribution. In this article, we investigate the issue of automatically detecting rumours as they circulate on social media. We provide a novel approach for automatically detecting rumours on social media that combines word embeddings with machine learning. The following listing summarises the important contributions of this work:

1. To address the cross-topic problems in broadcast rumour identification, we provide a novel technique for updating word embeddings while training proceeds using word2vec.

2. The experiment takes place in five different events from the PHEME dataset having conversational structure, and it is evaluated using a variety of measures, including accuracy, precision, recall, and the AUC/ROC curve. The results show that the ensemble model outperforms better with high accuracy by comparing the four classifiers for overall events in the dataset.

The rest of the content is structured as follows: the prior effort in the area of rumour detection is described in segment 2, an overview of the proposed method is discussed in section 3, simulation results are discussed in section 4, and the paper is concluded with future research directions in section 5, which is followed by references used in this research work.

2. Related work. The amount and speed of user-generated information on social media make rumour detection imperative. The first and most essential step in locating unverified information circulating on social media is rumour detection. It serves as the foundation for all subsequent research. The outcome of future tasks can be indirectly improved by increasing this task's accuracy.

Regardless of the source's veracity and position as a verified source, social media enables the spread of information. Several studies have demonstrated automatic rumour identification in social media data.

The first technique was put out by authors [8] and begins with identifying "signal tweets," which are then sorted into several clusters, each of which stands for a rumour. Next, further similar tweets are found using the summary of each cluster. The clusters are evaluated in order of how likely they will be – rumours at the end. The whole foundation of the suggested framework is a set of user-defined regular expressions. This list must be continuously edited and updated for the model to handle new, unseen tales more effectively. Another study in this category is provided in [9]. The researcher suggested a rumour identification model based on a sequential classifier, in which the text was classified as likely a rumour or a non-rumour based on preliminary data. While this technique outperforms the earlier efforts, it is affected by the cold start problem, which means the suggested sequential classifier's effectiveness depends on the tweets' structures observed thus far. Our suggested strategy addresses this issue by categorizing each micro-post solely on its features. It is not required to use micro-post procedures.

In paper [10] the authors suggested a rumour detection approach that matches social media material to news media and employs the support vector machine (SVM) as a binary classification strategy. However, it is not suitable for large amounts of data. In study [11] the authors suggested a combined approach for rumour classification that includes deep learning and an optimized filter-wrapper classifier for naive Bayes.

Articles of news that have been purposefully written to present incorrect information are called rumours. Recently, rumour detection has gained much interest as a developing study area, particularly on social media. This field of study aims to predict whether a news item is a rumour. The work in this sector may be divided into two families: selecting news stories worth checking and assessing their authenticity. In paper [12] the authors suggested automated categorization of news stories. Their study focuses on different textual properties that can be used to discern between genuine and phoney content. Using these properties, they trained various machine learning algorithms tested on four real-world datasets using different ensemble approaches. However, it was more expensive to create, train, and deploy. In study [13] the authors focused on rumour detection using NLP and data mining techniques. Authors categorize fake news that spread on social media into two categories: rumours that have been around for a while and newly developing rumours that result from current occurrences. Authors create a four-part framework for categorizing rumours: 1) rumour detection; 2) rumour tracking; 3) rumour stances; 4) rumour veracity.

Data imputation was used to improve the identification of fake news, according to the authors [14]. To improve performance, the authors used a novel data pre-processing method to fill in the absent value in the original dataset. Using data modelling, the authors used missing numbers for numerical and hierarchical traits. When authors selected the most common value in columns and were numerical for the average value of the column, the accuracy of multilayer perceptron (MLP) classes were shown to be improved for hierarchies.

Study [15] provided several important influencing factors, and the innovative rumour-refuting method presented in this study used NLP to analyze user microblog content before using the NLP analysis findings as extra predictor factors to identify anti-rumour spreaders. This research provides a novel and efficient decision support system for rumour countermeasures compared to prior studies. The number of pairwise similarities increases quadratically with dataset size, posing a problem for similarity-based machine learning techniques on big datasets. As a result, computing all potential pairwise similarities for big databases is virtually impossible.

3. Rumour Detection on social media by Machine Learning and Deep Learning Models. Any information, whether real or fake, has the potential to become viral on social media and touch millions of people. Even rumours spread quickly because of how quickly information is shared. As a result, it's critical to identify and suppress these rumours before they seriously affect people's lives.

The goal is to determine whether a specific micro-post on a particular bit of data is a story. It is the definition of the research issue of detecting breaking news rumours. This issue may be expressed as a binary classification problem in the manner described below: Let the word sequence $w = \{wc_1 \ldots wc_N\}$ represent a micro-post w of length N. The objective is to determine if w is a rumour or non-rumour by selecting a label from the set $S = \{R, NR\}$.

Figure 2 shows the suggested rumour detection model's workflow. There are four stages in the model: 1) data collection; 2) pre-processing; 3) feature extraction; 4) training and testing of Data using a machine learning algorithm. Accuracy, precision, recall, and F1-Score measurements were employed to evaluate the effectiveness of each technique. More details on the steps are provided in the following subsections.

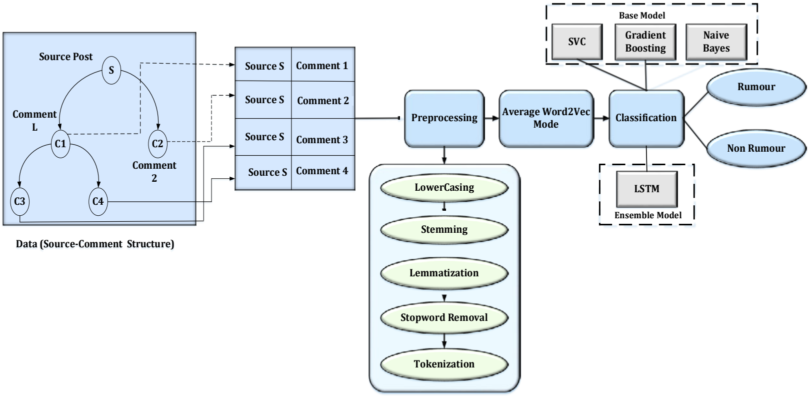

Fig. 2. Workflow of Proposed Methodology

3.1. Data Collection. The standard PHEME dataset utilized in this study for rumour identification [16] includes tweets about five significant news events that seasoned journalists have marked for the 'rumour' and 'non-rumour' classifications. Here, the objective is to simulate how a source post's comments behave in a conversational structure. The whole network was processed by taking into account the source node and their immediate comment reply node. All source posts and immediate nodes were then saved in CSV format. These are the events:

1. #charliehebdo – around noon on January 7, 2015, two gunmen attacked the Paris headquarters of the French satirical weekly newspaper Charlie Hebdo, murdering 12 persons and injuring 11. The collection comprehends 458 whispers and 1621 non-rumours.

2. #ferguson – on August 9, 2014, Darren Wilson, a white police officer, mortally shot Michael Brown Jr., an 18-year-old African American, in Ferguson, Missouri. The officer describes an encounter with Brown during which Brown assaulted him. Wilson later pursued him after he fled. Wilson fired twelve shots, six of which struck Brown from the front. Numerous demonstrations ensued. There are 284 rumours and 859 non-rumours in the dataset.

3. #germanwingscrash – on March 24, 2015, an Airbus A320-211 transporting 144 people and six crew members crashed in the French Alps, 100 kilometres (62 miles) northwest of Nice, while en transit from Barcelona-El Prat Airport in Spain to Düsseldorf Airport in Germany. The co-pilot, whom his psychiatrist had identified as having suicidal inclinations, deliberately crashed the aircraft. The collection contains 238 whispers and 231 non-rumours.

4. #ottawashooting – on October 22, 2014, many gunshots happened on Parliament Hill in Ottawa. Michael Zehaf-Bibeau mortally shot Corporal Nathan Cirillo, a Canadian soldier acting as a symbolic sentry at the Canadian National War Memorial. Zehaf-Bibeau then went inside the adjoining Centre Block of the Canadian Parliament, where MPs were conducting caucuses. After a struggle with a policeman outside, Zehaf-Bibeau rushed inside and engaged in a firefight with parliament security. Six cops shot him 31 times, and he passed away there and then. The dataset comprises 470 rumours and 420 non-rumours.

5. #sydneysiege – on the 15th and 16th of December, 2014, a gunman named Man HaronMonis kidnapped ten clients and eight employees at a Lindt confectionery café in Sydney, Australia. Two individuals were murdered, and a couple more were injured during the 16-hour standoff.

One indicates a rumour, and zero indicates a non-rumour. The data cleaning technique is applied in two different processes. The deactivated ID must be eliminated once the null value has been eliminated.

3.2. Pre-processing. The process of changing data so that the machine learning program can understand is known as pre-processing. The model is fed with clean data transformed from the original raw data. Especially for social media content, data pre-processing is an important stage. These datasets must go through the following rounds of pre-processing to receive specific refinements. The crucial cleaning procedures within the pre-processing techniques applied in this study include stop-word removal, tokenization, lowercase, deleting repeated letters, auto-correct spelling, and stemming. They will assist us by removing redundant data and reducing the actual data's quantity. Then, to achieve better execution, the pre-processing entails the ordering of the processes stated in the following steps:

1. Getting rid of the unnecessary: Punctuation indicators such as commas, apostrophes, quotation marks, and question marks are deleted if they contribute no value to the performance.

2. Elimination of Stop Words: Stop words are frequently used to describe the most general terms in a language that lack substantial semantic substance in an assertion. Each tweet in the datasets has these terms removed.

3. Lowercasing is changing uppercase characters to lowercase letters in a phrase. For example, "Charlie Hebdo became well known for publishing the Muhammed cartoons two years ago" will be converted into "charlie hebdo became well known for publishing Muhammed cartoons two years ago".

4. Tokenization is the process of reducing vast quantities of substance into smaller token-sized bits. The tokens, split down into words, are counting the sentences. For example, the processed input tokenizes such as, 'charlie', 'hebdo', 'became', 'well', 'known', 'publishing', 'Muhammed', 'cartoon', 'two', 'year', 'ago'.

5. The stemming and Lemmatization process recovers the lemma form for each input word, and end users can ask any form of a base word and receive relevant results, reducing the word to its word stem.

The pre-processed tweets are fed to the feature extraction stage to extract the features to classify the rumour and non-rumour contents.

3.3. Feature Extraction using Word2Vec. A word2vec model uses a text dataset as input to generate real-valued, low-dimensional vector approximations of the words in the corpus. As a result, written information is transformed into dispersed vector representations, which may subsequently be used as input for the classifier to classify rumours and non-rumours. Word embedding is vector depictions of words.

In this research, we use the Google News Dataset [17] to train the word2vec model more efficiently using a method dubbed skip-gram. The word2vec model [18] is created using skip-gram using a text corpus. Let wc_i be a word in the corpus, and let the context of wc_i be the collection of words surrounding wc_i inside a certain window size of a phrase. By learning the word representations of each word wc_i and the words in their context, skip-gram constructs the word2vec model. The learning goal in this instance is to find meaningful representations of these phrases in the embedding space so that the model, given any other word, wt, can predict its surrounding context words with high probabilities while the others have low probabilities. A skip-gram model's formal goal is to maximize the average log probability function shown shown in equation (1), given a list of words $w = \{wc_1 ... wc_N\}$ and a context window of size W.

$$Average \log probablity\ function = \frac{1}{N}\sum_{n=1}^{N}\sum_{-w\le j\le w, j\ne 0} \log p(wc_{n+j} \mid wc_n)\ , \qquad (1)$$

where $\log p(wc_{n+j} wc_n)$ is approximated using negative sampling. The Word2vec probabilistic model is used to anticipate the context of the nearby words. The words were predicted using a mix of model architectures depending on the corresponding context word. The dataset contains a substantial quantity of data and is transformed into word vectors. It is passed into the classifiers.

3.4. Applying learning models. We tested the efficacy of different models using three conventional machine-learning classifiers. Three different models were tested to choose the most effective model: support vector machine (SVM), Naive Bayes (NB) and gradient boosting classifier.

3.4.1. Support Vector Classifier (SVC). SVM [19], a subset of machine learning, seeks the optimal hyperplane, which can often divide data sets into two groups. A neural network approach merely looks for hyperplane separators between classes, whereas SVM seeks the optimal hyperplane. It is said to be the best if a hyperplane has the widest margin and can categorize fresh data effectively and without mistakes – a hyperplane in the shape of a straight line for two-dimensional data. SVM is widely regarded as effective in real-world applications where data sets are usually segregated nonlinearly and provide superior results than other methods. In non-linear SVM, data is transferred to an advanced dimensional vector space by the function ϕ (x) and then a hyperplane can be created to serve as a class separator.

The dot product produced from the converted data, namely ϕ_x, ϕ_y, is the sole component of the SVM learning process dependent on it. Because transformation is challenging to convert, kernel K (z_x, zy) can be used instead of computing dot products. The following formula denotes this function as Kernel Trick:

$$K\ \left(z_x,\ zy\right)\ =\ \phi_x . \phi_y . \qquad (2)$$

By knowing the kernel functions that will be used, the Kernel Trick can streamline identifying support vectors. In our work, we have used RBF (radial Bias Function) has been used as a kernel function.

3.4.2. Gradient Boosting Classifier. A strong classifier is gradually built from several weak learners using the additive and iterative tree-based supervised machine learning technique called "boosting" [20]. Gradient boosting creates a prediction model from a collection of poor prediction models, most commonly decision trees. When a decision tree fails to learn, the technique that results is known as gradient-enhanced trees, and it typically beats random forest. It builds the model in the same way that other boosting techniques do, but it broadens the scope by enabling the optimization of any differentiable loss function.

3.4.3. Naive Bayes Classifier. The naive Bayes technique is founded on the Bayes hypothesis [21]. It employs training data to calculate the likelihood that certain attributes belong to a certain class to forecast the unobserved data class (testing data).

The posterior probability, P(a|z), can be calculated using the Bayes theorem by dividing the P(z) with the product of P(a) and P(z|a). The naive Bayes classifier assumes that the impact of a predictor's value (z) on a certain class (a) is unrelated to the values of other predictors. The term "class conditional independence" refers to this assumption:

$$P(a|z) = \frac{P(z \mid a)P(a)}{P(z)}, \tag{3}$$

$$P(a|Z) = P(z1|a) \times P(z2|a) \times \ldots\ldots P(z2 \mid a) \times P(a), \tag{4}$$

where:

P(a|z) is the posterior probability of the class-given attribute.

P(a) is the prior probability of class.

P(z|a) is the likelihood which is the predictor given class probability.

P(z) is the prior probability of the predictor.

The proposed model consists of three machine learning base models: Support Vector Classifier (SVC), Gradient Boosting, and Naive Bayes. These models are combined with an ensemble model called Long Short-Term Memory (LSTM).

3.4.4. Short-Term Long Memory (LSTM). A modified RNN network called long short-term memory (LSTM) was developed to learn long-range relationships among time-varying patterns [19]. In general, LSTM is a second-order recurrent neural network that addresses the problem of disappearing and exploding gradients by substituting memory blocks in the recurrent hidden layer for RNN simple units. In an LSTM, a memory block is a complicated multi-unit processing unit. It comprises one or more memory cells, input, output, and forget adaptive multiplicative gating units, and a self-recurrent connection with a fixed weight of 1.0. With help from adaptive multiplicative gating units, it functions as a short-term memory. Input and output gates, respectively, govern a memory cell's input and output flow during activation.

In general, an Android app features $y = (y_1, y_2, . . ., y_T)$ are transmitted as input to LSTM, which then estimates an output sequence $zt = (zt_1, zt_2, ..., zt_T)$ with a continuous estimation of the gates that include input gate (zg), output gate (xg), forget gate (fg), and enhancing a memory cell (ml) activations iteratively from t = 1 to T. It is possible to formulate the calculation of the recurrent hidden layer function at time-step T as follows:

$$y_t, hz_{t-1}, ml_{t-1} \to hz_t, ml_t, \tag{5}$$

$$zg_t = sigm\left(\omega_{yzg} y_t + \omega_{hzzg} hz_{t-1} + \omega_{mlzg} ml_{t-1} + a_{zg}\right), \tag{6}$$

$$fg_t = sigm\left(\omega_{yfg} y_t + w_{hzfg} hz_{t-1} + \omega_{fgml} ml_{t-1} + a_{fg}\right), \tag{7}$$

$$ml_t = fg_t \odot ml_{t-1} + zn_t \odot \tanh\left(\omega_{mly} y_t + \omega_{mlhz} hz_{t-1} + a_{ml}\right), \tag{8}$$

$$xt_n = sigm\left(\omega_{xty} y_t + \omega_{xthi} hi_{t-1} + \omega_{xtml} ml_t + a_{xt}\right), \tag{9}$$

$$hz_t = xt_t \odot \tanh\left(ml_t\right), \tag{10}$$

where tanh represents the element-wise hyperbolic tangent non-linear activation function with outcomes in a range of [-1,1], sigm is known as element-wise sigmoid non-linear activation function with an outcome range of [0, 1], $\odot$ stands for element-wise vector multiplication, "ω terms" represents the weight matrices, "a_{in}," "a_{fg}," "a_{ml}," and "a_{ot}" are bias units of input gate, forget gate, memory cell, and output gate, correspondingly and ml_t instructed LSTM network to acquire knowledge of long-term temporal dynamics in the sequence data through proactively selecting either the current input value employing input gate zg_t or forgetting earlier stated data with f g_t.

4. Result and Discussion. A developed model's performance was assessed for individual events in the PHEME dataset, considering the features from the conversation comments for the source post. Experiments were conducted on a typical computer. (Intel Core i5-6500 3.20 GHz, 8 GB RAM). Python3.11.1 with Jupyter Notebook was used to create the model. The pandas, nltk, genism for word2vec, sklearn for machine learning model, and NumPy libraries are included in the Python packages. The classifier used the percentage method (80% training and 20% testing datasets) to analyze performances.

4.1. Performance Evaluation. To accurately depict performance on the PHEME dataset, we selected Accuracy, Precision, and Recall as assessment measures for individual events. We also evaluated our model's performance using AUC. The AUC scores for the PHEME dataset's five events vary from 0.78 to 0.91 (Table 1).

Table 1. Events Wise performance evaluation

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	AUC /ROC (%)
Charlie Hebdo				
SVC	93.60	97	95	87.2
Gradient Boosting Classifier	92.33	85.0	88	84.8
Naïve Bayes	82.73	80	72	79.8
Ensemble Model	93.75	97.62	95.38	92.13
Ferguson				
SVC	93.40	98	93	89.8
Gradient Boosting Classifier	92.14	97	92	88
Naïve Bayes	77.85	81	88	75.4
Ensemble Model	93.85	98.62	95.38	92.13
Ottawa shooting				
SVC	86.92	91	84	87
Gradient Boosting Classifier	91.48	93	90	91.5
Naïve Bayes	78.76	79	77	78.8
Ensemble Model	92.41	94.02	91.44	91.98
Sydney siege				
SVC	93.50	98	94	90
Gradient Boosting Classifier	92.25	98	92	88
Naïve Bayes	78.53	83	87	75.3
Ensemble Model	94.41	98.22	94.44	91.48
German wings crash				
SVC	93.16	98	93	89.5
Gradient Boosting Classifier	92.25	97	92	88.2
Naïve Bayes	78.42	82	87	75.6
Ensemble Model	93.55	98.62	95.38	92.13

Table 1 shows the results of each unique event in PHEME investigated using different classifiers such as SVC, NB, and the gradient boosting method along with the ensemble model. The ensemble model outperforms the other classifiers for the five events, with the highest accuracy for the Sydney siege.

For the Charlie Hebdo, Ferguson Unrest events, Ottawa shooting, Sydney siege, and German wings crash, the ensemble model has an accuracy of 93.75%, 93.85%, 92.41%, 94.41%, and 93.55, respectively.

By comparing the results for four classifiers for combined overall events in the datasets in positions of performance metrics like accuracy, precision, recall, and AUC/ROC curve, the ensemble model outperformed better than other classifiers. In Table 2 the three classifiers are compared with the final dataset in terms of performance metrics, which results that the ensemble model has the highest values of accuracy is 93.59%, precision is 97.42%, recall is 94.40%, and AUC/ROC is 91.97% due to its linear function of predicting the rumour data from non-rumour data accurately.

Table 2. Performance of Different classifiers on overall events in the PHEME dataset

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	AUC /ROC (%)
SVC	93.19	96	94	91
Gradient Boosting Classifier	85.25	92	88	80.8
Naïve Bayes	77.52	80	87	77.52
Ensemble Model	93.59	97.42	94.40	91.97

5. Conclusion. In times of crisis, rumours are commonplace. Rumours grow swiftly in the virtual social environment due to the intensity and unpredictability of the issue, as well as the informational gap. The tangibility of information must thus always be called into question. This study examined a unique system for real-time rumour detection to dispel online rumours. The classification is carried out using three different supervised learning methods: SVC, Gradient Boosting, and Naïve Bayes classifiers, and are combined with an ensemble model called Long Short-Term Memory (LSTM) to overcome the drawbacks of the three classifiers. The classification takes place for each classifier by using five different events in the PHEME dataset. The performance of each classifier for five events is calculated using several performance measures like accuracy, precision, recall, and AUC/ROC curve. The results show that the ensemble model outperformed other classifiers by comparing the performance of all events in the datasets for three different classifiers. This study may be expanded by employing various deep-learning models to increase the prediction accuracy of rumour detection. We can also increase forecast accuracy by applying efficient pre-processing processes.

References

1. Bingol H., Alatas B. Rumor Detection in Social Media using machine learning methods. 1st International Informatics and Software Engineering Conference (UBMYK). 2019. pp. 1–4.
2. He Z., Li C., Zhou F., Yang Y. Rumor detection on social media with event augmentations. Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021. pp. 2020–2024.
3. Pathak A.R., Mahajan A., Singh K., Patil A., Nair A. Analysis of techniques for rumor detection in social media. Procedia Computer Science. 2020. vol. 167. pp. 2286–2296.
4. Xuan K., Xia R. Rumor stance classification via machine learning with text, user and propagation features. International Conference on Data Mining Workshops (ICDMW). 2019. pp. 560–566.
5. Shi A., Qu Z., Jia Q., Lyu C. Rumor detection of COVID-19 pandemic on online social networks. IEEE/ACM Symposium on Edge Computing (SEC). 2020. pp. 376–381.
6. Gidwani M., Rao A. Early Identification of Rumors in Social Media: A Survey. Modern Approaches in Machine Learning & Cognitive Science: A Walkthrough. 2022. pp. 305-316. DOI: 10.1007/978-3-030-96634-8_28.
7. Hassan N.Y., Gomaa W.H., Khoriba G.A., Haggag M.H. Supervised learning approach for Twitter credibility detection. 13th International Conference on computer engineering and Systems (ICCES). 2018. pp. 196–201.
8. Zhao Z., Resnick P., Mei Q. Enquiring minds: Early detection of rumors in social media from enquiry posts. Proceedings of the 24th international conference on world wide web. 2015. pp. 1395–1405.
9. Zubiaga A., Liakata M., Procter R. Learning reporting dynamics during breaking news for rumour detection in social media. Computer Science. arXiv preprint arXiv:1610.07363. 2016. 20 p.
10. Kandasamy N., Murugasamy K. Detecting and filtering rumor in social media using news media event. Concurrency and Computation: Practice and Experience. 2021. vol. 33(5). DOI:10.1002/cpe.6329.
11. Kumar A., Bhatia M.P.S., Sangwan S.R. Rumour detection using deep learning and filter-wrapper feature selection in benchmark Twitter dataset. Multimedia Tools and Applications. 2022. vol. 81. no. 24. pp. 34615–34632.
12. Ahmad I., Yousaf M., Yousaf S., Ahmad M.O. Fake news detection using machine learning ensemble methods. Complexity. 2020. vol. 2020. pp. 1–11.
13. Zubiaga A., Aker A., Bontcheva K., Liakata M., Procter R. Detection and resolution of rumours in social media: A survey. ACM Computing Surveys (CSUR). 2018. vol. 51. no. 2. pp. 1–36.
14. Kotteti C.M.M., Dong X., Li N., Qian L. Fake news detection enhancement with data imputation. 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.
15. Li Z., Zhang Q., Du X., Ma Y., Wang S. Social media rumor refutation effectiveness: Evaluation, modelling and enhancement. Information Processing & Management. 2021. vol. 58(1), 102420.
16. Zubiaga A., Wong Sak Hoi G., Liakata M., Procter R. PHEME dataset of rumours and non-rumours. 2016. DOI: 10.6084/M9.FIGSHARE.4010619.V1.
17. Karkali M., Rousseau F., Ntoulas A., Vazirgiannis M. Efficient online novelty detection in news streams. Proceedings of International Conference on Web Information Systems Engineering. 2013. pp. 57–71. DOI: 10.1007/978-3-642-41230-1_5.
18. Vijayaraghavan S., Wang Y., Guo Z., Voong J., Xu W., Nasseri, A., Cai J., Li L., Vuong K., Wadhwa E. Fake news detection with different models. arXiv preprint arXiv:2003.04978. 2020. DOI: 10.48550/arXiv.2003.04978.
19. Vinothkumar S., Varadhaganapathy S., Ramalingam M., Ramkishore D., Rithik S., Tharanies K.P. Fake News Detection Using SVM Algorithm in Machine Learning. Proceedings of International Conference on Computer Communication and Informatics (ICCCI). 2022. pp. 1–7. DOI: 10.1109/ICCCI54379.2022.9740886.
20. Birunda S.S., Devi R.K. A novel score-based multi-source fake news detection using gradient boosting algorithm. Proceedings of International Conference on Artificial Intelligence and Smart Systems (ICAIS). 2021. pp. 406–414.
21. Granik M., Mesyura V. Fake news detection using naive Bayes classifier. Proceedings of IEEE first Ukraine conference on electrical and computer engineering (UKRCON). 2017. pp. 900–903.

Gidwani Manya — Research Scholar, Department of information technology, Mukesh Patel School of Technology Management and Engineering (MPSTME), Shri Vile Parle Kelavani Mandal's Narsee Monjee Institute of Management Studies (SVKM's NMIMS). Research interests: machine learning, NLP, deep learning, social media analysis. The number of publications — 7. renu.meghani@gmail.com; V.L. Mehta Road, Vile Parle, West, Maharashtra, Mumbai, India; office phone: +91(224)233-4000.

Rao Ashwini — Ph.D., Dr.Sci., Assistant professor of the department, Department of information technology, Mukesh Patel School of Technology Management and Engineering (MPSTME), Shri Vile Parle Kelavani Mandal's Narsee Monjee Institute of Management Studies (SVKM's NMIMS). Research interests: machine learning, NLP, social media analysis. The number of publications — 13. ashwini.rao@nmims.edu; V.L. Mehta Road, Vile Parle, West, Maharashtra, Mumbai, India; office phone: +91(224)233-4000.

DOI 10.15622/ia.22.4.3

М. Гидвани, А. Рао

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ВЫЯВЛЕНИЯ СЛУХОВ В СОЦИАЛЬНЫХ СЕТЯХ С ИСПОЛЬЗОВАНИЕМ РАЗЛИЧНЫХ КЛАССИФИКАТОРОВ

Гидвани М., Рао А. **Сравнительный анализ выявления слухов в социальных сетях с использованием различных классификаторов.**

Аннотация. По мере того, как число пользователей социальных сетей растет, создание и распространение информации увеличиваются каждый день в огромных масштабах. Люди могут делиться своими идеями и мнениями на этих платформах. Сайт микроблогов в социальных сетях, таких как Facebook или Twitter, является предпочтительным средством обсуждения любого важного события, и информация передается немедленно. Это приводит к быстрому распространению слухов и распространению неточной информации, что вызывает у людей беспокойство. Поэтому важно оценить и подтвердить уровень достоверности такой информации. Из-за сложности текста автоматическое обнаружение слухов на ранних стадиях затруднительно. В данном исследовании используются различные методы NLP для извлечения информации из твитов, а затем применяются различные модели машинного обучения, чтобы определить, является ли информация слухом. Классификация выполняется с использованием трех классификаторов, таких как SVC (Support Vector Classifier), Gradient Boosting и классификаторы Naive Bayes для пяти различных событий из набора данных PHEME. Существуют некоторые недостатки: ограниченная обработка несбалансированных данных, трудность улавливания сложных лингвистических шаблонов, отсутствие интерпретируемости, сложности с обработкой больших пространств признаков и нечувствительность к порядку слов и контексту при использовании вышеуказанных классификаторов. Подход суммирования используется для преодоления вышеуказанных недостатков, при котором выходные данные комбинированных классификаторов представляют собой ансамбль с LSTM. Была проанализирована производительность моделей. Экспериментальные результаты показывают, что ансамблевая модель дает эффективные результаты по сравнению с другими классификаторами с точностью 93,59%.

Ключевые слова: обнаружение слухов, машинное обучение, социальные сети, SVC, градиентный бустинг, наивный байесовский классификатор.

Литература

1. Bingol H., Alatas B. Rumor Detection in Social Media using machine learning methods. 1st International Informatics and Software Engineering Conference (UBMYK). 2019. pp. 1–4.
2. He Z., Li C., Zhou F., Yang Y. Rumor detection on social media with event augmentations. Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021. pp. 2020–2024.
3. Pathak A.R., Mahajan A., Singh K., Patil A., Nair A. Analysis of techniques for rumor detection in social media. Procedia Computer Science. 2020. vol. 167. pp. 2286–2296.
4. Xuan K., Xia R. Rumor stance classification via machine learning with text, user and propagation features. International Conference on Data Mining Workshops (ICDMW). 2019. pp. 560–566.

5. Shi A., Qu Z., Jia Q., Lyu C. Rumor detection of COVID-19 pandemic on online social networks. IEEE/ACM Symposium on Edge Computing (SEC). 2020. pp. 376–381.
6. Gidwani M., Rao A. Early Identification of Rumors in Social Media: A Survey. Modern Approaches in Machine Learning & Cognitive Science: A Walkthrough. 2022. pp. 305-316. DOI: 10.1007/978-3-030-96634-8_28.
7. Hassan N.Y., Gomaa W.H., Khoriba G.A., Haggag M.H. Supervised learning approach for Twitter credibility detection. 13th International Conference on computer engineering and Systems (ICCES). 2018. pp. 196–201.
8. Zhao Z., Resnick P., Mei Q. Enquiring minds: Early detection of rumors in social media from enquiry posts. Proceedings of the 24th international conference on world wide web. 2015. pp. 1395–1405.
9. Zubiaga A., Liakata M., Procter R. Learning reporting dynamics during breaking news for rumour detection in social media. Computer Science. arXiv preprint arXiv:1610.07363. 2016. 20 p.
10. Kandasamy N., Murugasamy K. Detecting and filtering rumor in social media using news media event. Concurrency and Computation: Practice and Experience. 2021. vol. 33(5). DOI:10.1002/cpe.6329.
11. Kumar A., Bhatia M.P.S., Sangwan S.R. Rumour detection using deep learning and filter-wrapper feature selection in benchmark Twitter dataset. Multimedia Tools and Applications. 2022. vol. 81. no. 24. pp. 34615–34632.
12. Ahmad I., Yousaf M., Yousaf S., Ahmad M.O. Fake news detection using machine learning ensemble methods. Complexity. 2020. vol. 2020. pp. 1–11.
13. Zubiaga A., Aker A., Bontcheva K., Liakata M., Procter R. Detection and resolution of rumours in social media: A survey. ACM Computing Surveys (CSUR). 2018. vol. 51. no. 2. pp. 1–36.
14. Kotteti C.M.M., Dong X., Li N., Qian L. Fake news detection enhancement with data imputation. 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.
15. Li Z., Zhang Q., Du X., Ma Y., Wang S. Social media rumor refutation effectiveness: Evaluation, modelling and enhancement. Information Processing & Management. 2021. vol. 58(1), 102420.
16. Zubiaga A., Wong Sak Hoi G., Liakata M., Procter R. PHEME dataset of rumours and non-rumours. 2016. DOI: 10.6084/M9.FIGSHARE.4010619.V1.
17. Karkali M., Rousseau F., Ntoulas A., Vazirgiannis M. Efficient online novelty detection in news streams. Proceedings of International Conference on Web Information Systems Engineering. 2013. pp. 57–71. DOI: 10.1007/978-3-642-41230-1_5.
18. Vijayaraghavan S., Wang Y., Guo Z., Voong J., Xu W., Nasseri, A., Cai J., Li L., Vuong K., Wadhwa E. Fake news detection with different models. arXiv preprint arXiv:2003.04978. 2020. DOI: 10.48550/arXiv.2003.04978.
19. Vinothkumar S., Varadhaganapathy S., Ramalingam M., Ramkishore D., Rithik S., Tharanies K.P. Fake News Detection Using SVM Algorithm in Machine Learning. Proceedings of International Conference on Computer Communication and Informatics (ICCCI). 2022. pp. 1–7. DOI: 10.1109/ICCCI54379.2022.9740886.
20. Birunda S.S., Devi R.K. A novel score-based multi-source fake news detection using gradient boosting algorithm. Proceedings of International Conference on Artificial Intelligence and Smart Systems (ICAIS). 2021. pp. 406–414.

21. Granik M., Mesyura V. Fake news detection using naive Bayes classifier. Proceedings of IEEE first Ukraine conference on electrical and computer engineering (UKRCON). 2017. pp. 900–903.

Гидвани Манья — ученый-исследователь, кафедра информационных технологий, Школа технологического менеджмента и инженерии имени Мукеша Пателя, Институт управленческих исследований Нарси Монджи Шри Вайла Парле Келавани Мандала. Область научных интересов: машинное обучение, НЛП, глубокое обучение, анализ социальных сетей. Число научных публикаций — 7. renu.meghani@gmail.com; В.Л. Мехта-роуд, Виле Парле, Запад, Махараштра, Мумбаи, Индия; р.т.: +91(224)233-4000.

Рао Ашвини — Ph.D., Dr.Sci., доцент кафедры, кафедра информационных технологий, Школа технологического менеджмента и инженерии имени Мукеша Пателя, Институт управленческих исследований Нарси Монджи Шри Вайла Парле Келавани Мандала. Область научных интересов: машинное обучение, НЛП, анализ социальных сетей. Число научных публикаций — 13. ashwini.rao@nmims.edu; В.Л. Мехта-роуд, Виле Парле, Запад, Махараштра, Мумбаи, Индия; р.т.: +91(224)233-4000.