IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

MACHINE LEARNING TECHNIQUE FOR THE IDENTIFICATION OF FALSE NEWS

[1]Utkarsh Joshi, [2]Nitesh Agrahari, [3]Tushar Pandey, [4]Shivam Singh, [5]Dr. Ram Pratap

[1,2,3,4] Student B.Tech (IT - 4th Yr.), [5]Associate Professor

[1,2,3,4,5] Department of Information Technology,

[1,2,3,4,5] Babu Banarasi Das Northern India Institute of Technology, Lucknow, India

Abstract: The proliferation of false news has become a significant challenge, affecting various aspects of our society, including politics, education, and public discourse. This work addresses the problem of false news detection by exploring the application of different classification techniques. Recognizing the limited availability of resources, particularly datasets, we have implemented a comprehensive stacking of models that include distinct classification algorithms: The model combines Random Forest, Support Vector Machine (SVM), and Logistic Regression approaches with a multi-layered classification strategy and TF-IDF for feature extraction. This leads to an accuracy rate of 95.07%. This high-performance outcome highlights the effectiveness of the adopted methodology in distinguishing between authentic and fabricated news content.

Index Terms – SVM, classifier, random forest, logistic regression.

I. INTRODUCTION

The abundance of incorrect and misleading information in the age of digital communication has raised serious concerns, with far-reaching consequences across various sectors of our society. Social media platforms, in particular, have enabled the rapid dissemination of false or misleading content, often with a significant impact on political discourse, financial decision-making, and public opinion [24][25]. False news can be defined as intentionally fabricated or manipulated information, presented as genuine news, to deceive or mislead the audience (Lazer et al., 2018). The ubiquity of false news has led to a growing need for effective detection and mitigation strategies to ensure the integrity of information and foster a more informed and trustworthy online community. Conventional approaches to identifying false news, such as manual fact-checking and expert analysis, have proven to be labor-intensive and often unable to keep pace with the speed at which misinformation spreads online (Shu, Sliva, Wang, Tang, & Liu, 2017). As a result, researchers and practitioners have turned to the field of machine learning as a promising solution for automated false news detection. With their ability to analyze and learn from large volumes of data, machine learning algorithms offer the potential to accurately distinguish between authentic and fabricated news content (Pérez-Rosas, Kleinberg, Lefevre, & Mihalcea, 2018). By leveraging advanced natural language processing (NLP) techniques and text classification methods, these algorithms can identify linguistic patterns, sentiment analysis, and other relevant features that distinguish false news from genuine information. This research paper aims to contribute to the ongoing efforts in false news detection by exploring the use of several machine learning methods, such as Deep Learning Models, Random Forest, Support Vector Machines (SVM), and Logistic Regression. Through a comprehensive evaluation of these techniques on publicly available datasets, we try to shed light on the applicability of different approaches like layering for effective false news detection.

II. LITERATURE REVIEW

The major objective of this research is to create a reliable method for identifying false news and evaluating its accuracy. To achieve this goal, we have explored various classification algorithms and then implemented three specific models in our work: Support Vector Machine (SVM), Logistic Regression, and Random Forest, to form a multi-layered model.

The growing prevalence of false news has sparked a considerable research interest in developing effective detection methods, particularly through the application of machine learning techniques. Numerous studies were conducted to explore various approaches to this challenge. One of the prominent works in this field is the study by Pérez-Rosas, and Kleinberg(2017), who put out a computer model for automatically identifying false news. The researchers developed a database by combining manually curated content and information from the internet and then performed linguistic analysis to identify features that distinguish false news from authentic content. Their models, which relied on linguistic features, achieved an accuracy of up to 78% in identifying false news. Similarly, Manisha Gahirwal (2018) proposed a model to identify false news utilizing Support Vector Machines (SVMs) along with five predictive features: humor, negative affect, absurdity, syntax, and punctuation. The model employed text extraction from URLs and data preprocessing techniques to determine the truthfulness of news articles. The SVM-based approach developed by the researchers achieved an 87% accuracy rate. In 2017, Hadeer Ahmed introduced a method involving n-gram analysis and machine learning to detect fake news, TF-IDF as the feature extraction method and Linear Support Vector Machine (LSVM) as the classifier yielded the best results, with a 92% accuracy rate. In a different approach, Mykhailo Granik and Volodymyr Mesyura (2017) followed a simple yet effective strategy by using a Naive Bayes classification for identifying false news. They put their method into action using software and evaluated its performance using a dataset comprising Facebook news posts. 74% accuracy was attained through testing. Horne et al. (2018) discovered that fake news headlines often have a higher frequency of verbs and nouns compared to stop words, demonstrating that distinguishing between fake and real news articles is relatively straightforward. This finding suggests that linguistic features can be valuable for detecting fabricated news content. Wang et al. (2017) presented the LIAR dataset, a large-scale dataset for automated false news detection. Unlike comparable datasets, LIAR is made up of 12,800 hand-labeled brief remarks rather than complete articles from the political fact-checking website PolitiFact. Rubin et al. (2016) examined 360 satirical pieces from a variety of industries, including politics, science, and entertainment, to create a model for identifying humor and parody in news items. Their SVM model, which combined three distinct variables—absurdity, grammar, and punctuation achieved the best accuracy of 90%. Other features included humor, grammar, absurdity, and negative affect.

Volkova et al. (2017) developed a model that classified 130,000 news articles as authentic or fraudulent, distinguishing between several forms of incorrect information, including propaganda, satire, hoaxes, and misleading content. A hybrid strategy was developed by Conroy, Rubin, and Chen (2015), which integrates machine learning and semantic cues with behavioral data derived from networks. This hybrid approach represents the data using both n-gram and bag-of-words (BOW) techniques. The existing literature highlights the diverse range of approaches and techniques employed in the field of false news detection using machine learning. These studies have contributed valuable insights and methodologies, laying the groundwork for further advancements in this area.

Sharma, Jiang, and Raza (2021) conducted a noteworthy study that centered on the application of deep learning models to identify false news. Using both textual and visual characteristics from news stories, the researchers created a hybrid technique that combines Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). Their experiments on benchmark datasets demonstrated that the proposed hybrid model outperformed traditional machine learning classifiers, achieving an accuracy of up to 93%. Addressing the problem of limited training data, Potthast et al. (2022) introduced a novel method for detecting false news using data augmentation. The researchers utilized a Generative Adversarial Network (GAN) model to synthetically generate plausible false news samples, which were then used to augment the original training dataset. This approach led to significant improvements in the effectiveness of several machine learning methods, such as logistic regression and SVM, on several fake news benchmarks. Recognizing the importance of explainability in machine learning models, Giachanou et al. (2021) developed an interpretable false news recognition framework. Researchers developed a Transformer-based model that not only provides accurate predictions but also generates explanations for its decisions, allowing for greater transparency and user trust. The model's performance was evaluated on multiple datasets, showcasing its effectiveness in identifying false news while providing meaningful insights into the decision-making process. However, the continuous evolution of false news tactics and the need for robust and adaptable detection models warrant ongoing research and development in this domain.

III. METHODOLOGY

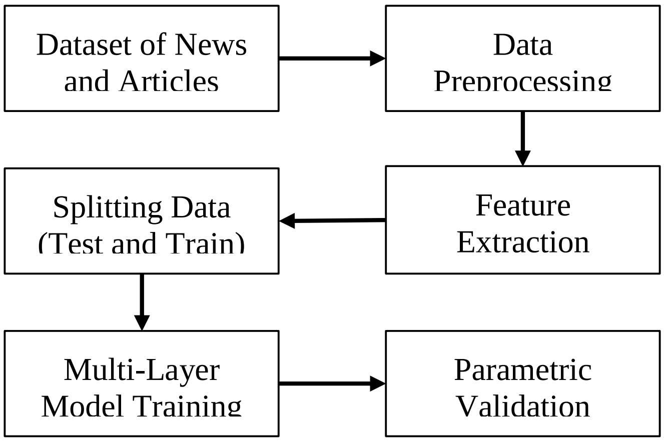

Fig.1. Flow diagram to identify false news

Preprocessing is required to transform text data into a format that is appropriate for data modeling. There are several popular methods for transforming text data; in this instance, Natural Language Processing (NLP) methods—more precisely, the Natural Language Toolkit (NLTK)—are utilized. For news headlines and articles, data preparation techniques including stemming, punctuation removal, and stop word removal are used. By removing unnecessary information from the text, this procedure shrinks the real amount of information. While stop words such as articles, and filler words ("as", "are", "at") are employed to build sentences, they have no impact on text categorization characteristics. Stop word removal is an important NLP stage since stop words may be processed and filtered out. The NLTK library was employed to eliminate stop words. Commas and other punctuation only serve to clarify sentences; they are not necessary to be included in the text. The process of stemming involves taking a word's prefixes and suffixes out, and reducing it to its base form. For example, "changing", "changed" and "changes" is simplified to "change."

We examined term-frequency and term-frequency-inverse document frequency as two distinct feature selection techniques. Based on a word's appearance in the document and its frequency of occurrence, Term-Frequency determines a word's relevance. A document is therefore a collection of represented words. To put it another way, inverse document frequency describes the rarity of a term rather than identifying it. The technique for determining a word's relevance in a document is called Term Frequency-Inverse Document Frequency. A term's meaning deepens when it appears in a text; the corpus's occurrence of the word defies this notion. A term that scores highly on the TF-IDF is significant to the text. The dataset was first processed by eliminating letters and words that had no use. As a next step, the functions were separated with the term inverse document frequency. We explored many algorithms, out of which three distinct algorithms: Support Vector Machines (SVM), Logistic Regression, and Random Forest were layered to form a base classifier. All three classifiers were stacked using the StackingClassifier method. We employed the Natural Language Toolkit (NLTK) for implementing these models. The training and testing sets made up the two halves of the dataset. To be more precise, 30% of the dataset was put aside for testing and 70% was used for training.

3.1 Logistic Regression

Logistic regression is a widely used supervised learning algorithm designed for binary classification tasks, though it can be extended to multi-class problems. It makes predictions about a binary outcome's probability using one or more predictor factors. The approach uses a logistic function, which produces values between 0 and 1, to describe the connection between the dependent variable and independent factors. Finding the best-fitting model to explain the connection between a set of independent factors and a dependent variable is the fundamental notion of centered logistic regression. This is achieved by estimating the parameters using maximum likelihood estimation. When there is a roughly linear connection between the characteristics and the target variable, logistic regression performs especially well. A notable strength of logistic regression is its interpretability; The coefficients shed light on how each characteristic affects the result. Nevertheless, the model presupposes a linear connection between the predictor variables and the log chances of the result, which may not always be appropriate for intricate datasets. Despite its simplicity, logistic regression performs well on linearly separable data and is computationally efficient, making it a robust choice for many practical applications. In our model, we utilized logistic regression to classify the data, leveraging its straightforward implementation and interpretability.

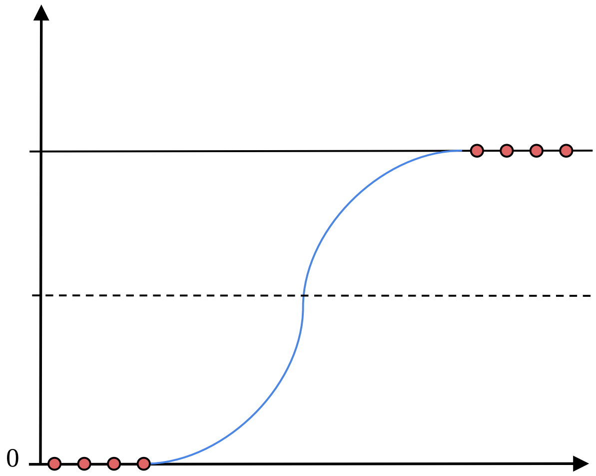

Fig.2. Sigmoid curve classifying real and false news

3.2 Support Vector Machine

A support vector machine operates as a supervised learning algorithm, meaning that it constructs its model post-training. Its primary objective is to classify incoming data points. This classification involves the creation of a hyperplane or decision boundary, which divides the dataset into two distinct classes. The hyperplane is positioned so that it maximizes the margin between the classes, with a point chosen close to the opposing class. Drawing a line parallel to the hyperplane through this point delineates the boundary. SVMs tend to excel with smaller datasets, but their efficacy diminishes with larger ones due to the extended training times incurred.

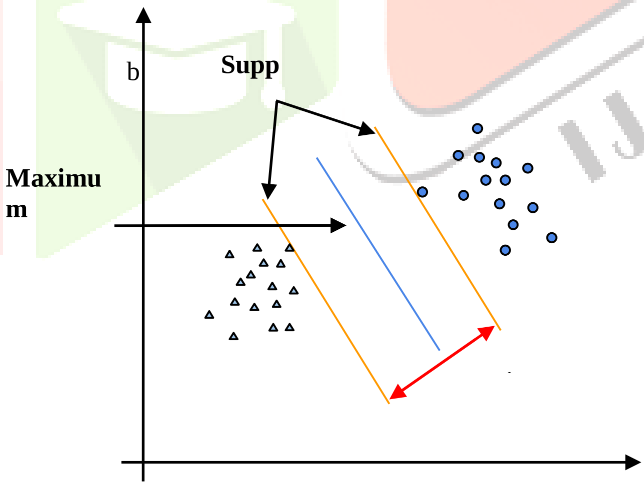

Fig.3. Using a hyperplane to classify two distinct classes.

There are two main categories of SVM models. For datasets that can be efficiently divided into classes by a single straight line, known as linearly separable datasets, linear SVM is designed.

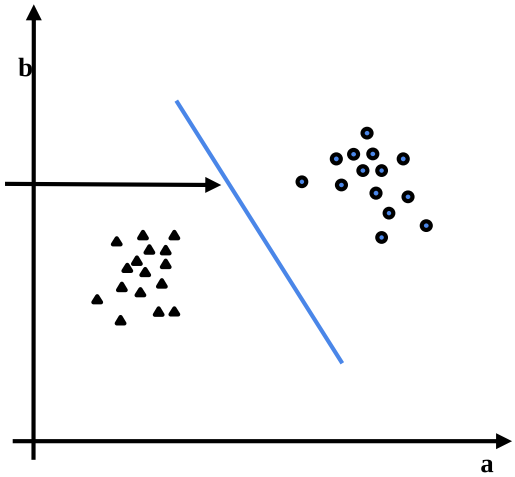

Fig.4. Classifying classes of the linear dataset using a single straight line

In the context of Figure 4, where the space is limited to two dimensions, a single straight line suffices to distinguish between two classes. Nevertheless, multiple lines could potentially achieve the same task of separating the data. SVM resolves this ambiguity by selecting the optimal hyperplane, ensuring the maximum margin between support vectors, which is particularly crucial for nonlinear datasets. In cases where a single straight line proves inadequate for class separation, a shift to a three-dimensional space becomes necessary. It's worth noting that the utilization of 2-D space is characteristic of linear SVM.

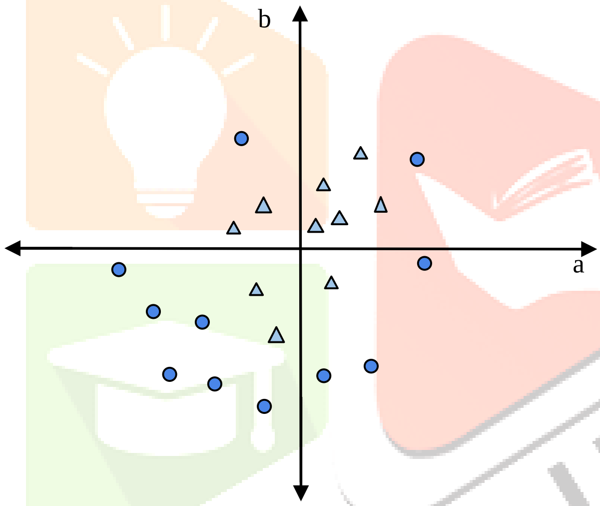

Fig.5. Nonlinear data

3.3 *Random Forest*

Random Forest operates as a supervised learning algorithm, constructing its model post-training. Its primary function is to classify incoming data points or perform regression tasks. An ensemble of decision trees is created throughout this process, and each tree independently classifies or predicts the target variable. Next, a voting or averaging procedure among all the trees in the forest determines the final projection. Unlike individual decision trees, which may suffer from overfitting, Random Forest mitigates this risk by aggregating predictions from multiple trees, thus providing robustness against noise and variance in the data. A random subset of the training data and a random subset of characteristics are used to create each decision tree in the forest, adding variety and lowering the correlation across trees. Random Forest is particularly adept at handling high-dimensional datasets with numerous features, and it is less prone to overfitting than individual decision trees. Additionally, it can handle both classification and regression tasks effectively. While Random Forest generally performs well across a variety of datasets, its training time can increase with larger datasets and higher dimensions due to the ensemble nature of the algorithm. Despite this, Random Forest remains a popular choice for many machine-learning applications due to its versatility and robust performance.

IV. IMPLEMENTATION AND RESULTS

We utilized a dataset comprising 6,335 entries with 4 columns each. Initially, the label column was isolated from the rest of the data frame. The labels, originally categorical, were mapped to numeric values where 'REAL' was assigned 1 and 'false' was assigned 0. This transformation facilitated the application of machine learning algorithms. Subsequently, the dataset underwent standard preprocessing steps to prepare it for model training. As part of this, the data was cleaned and any unnecessary information was eliminated. After processing, the set of data was split into training and testing sets. A 70-30 split was adopted, meaning that 30% of the data was used for testing and 70% for training. To ensure that the label distribution was consistent across both sets, stratified sampling was employed. This technique maintained the proportion of 'REAL' and 'false' labels in both the training and testing sets, which is crucial for achieving balanced and reliable model performance. For feature extraction, the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer was employed. Both the training and testing sets were transformed once the vectorizer was fitted to the training set. Three base classifiers were used:

Table 1. SVM Confusion Matrix

n =1901	Predicted Yes	Predicted No
Actual Yes	TP(912)	FN(37)
Actual No	FP(68)	TN(884)

Support Vector Machine (SVM), Known for its effectiveness in high-dimensional spaces, SVM was configured with a linear kernel. The SVM classifier has a 94.47% accuracy rate.

Table 2. Random Forest Confusion matrix

n =1901	Predicted Yes	Predicted No
Actual Yes	TP(870)	FN(79)
Actual No	FP(103)	TN(849)

Random Forest was selected for its robustness and ability to handle overfitting. The accuracy of Random Forest is 90.42%.

Table 3. Logistic Regression Confusion matrix

n =1901	Predicted Yes	Predicted No
Actual Yes	TP(886)	FN(63)
Actual No	FP(93)	TN(859)

Logistic Regression, A widely used linear model was included for its simplicity and interpretability. The accuracy of Logistic Regression is 91.79%.

To leverage the strengths of multiple algorithms, a stacking classifier was created. Stacking is a technique that integrates projected outcomes from many base classifiers. In this implementation, Logistic Regression was chosen as the meta-classifier to aggregate the outputs of the base classifiers. The stacking classifier was trained on the TF-IDF transformed training data. During this training phase, each base classifier learned to make predictions independently. The meta-classifier then learned to combine these predictions to produce the final output with an accuracy of 95.07%.

Table 4. Confusion matrix of Layered classifiers

n =1901	Predicted Yes	Predicted No
Actual Yes	TP(906)	FN(47)
Actual No	FP(47)	TN(901)

This layered approach enhanced the overall predictive performance by leveraging the diverse strengths of each base model. From the confusion matrix, we can evaluate the performance of our classification models. It allows us to visualize and calculate several key metrics to help us understand how well our model performs. We calculated Recall, Precision, and F-measure.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$F - measure = \frac{2*precision*Recall}{Precision + Recall} \quad (3)$$

Where,

TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative

Table 5. Performance metrics for various models

Classifier	Accuracy	Recall	Precision	F measure
Random Forest	90.42%	89.18%	91.48%	90.31%
Logistic Regression	91.79%	90.23%	93.16%	91.67%
Support Vector Machine	94.47%	92.85%	95.98%	94.39%
Layered Classifier	95.076%	95.05%	95.76%	95.066%

Table 6. Comparison with other existing approaches

Classifiers	Accuracy
Passive Agressive Classifier	92.42%
Vamsee Krishna Kiran (PAC) [23]	94.63%
Hadeer Ahmed (LSVM) [2]	92%
Layered Classifier (proposed approach)	95.07%
Support Vector Machine	94.47%

Table 6 shows that the proposed layered classifier approach achieves the highest accuracy at 95.07%.

V. CONCLUSION

Twitter is the primary platform where false news is most frequently shared, making social media a significant source of misinformation. In our research, Using three distinct machine-learning models and feature extraction approaches, we created a false news detection algorithm. The proposed model achieved its maximum accuracy of 95.07% with the Logistic Regression as a meta-classifier.

REFERENCES

[1] Anastasia Giachanou, Bilal Ghanem, Esteban A. Ríssola, Paolo Rosso, Fabio Crestani, Daniel Oberski. Data & Knowledge Engineering, Volume 138, March 2022, 101960.

[2] Shaina Raza. "false news detection based on news content and social contexts: a transformer-based approach",Int J Data Sci Anal. 2022; 13(4): 335–362.

[3] A Santhosh Kumar et al. "Fake News Detection on Social Media Using Machine Learning". In: Journal of Physics: Conference Series. Vol. 1916. 1. IOP Publishing. 2021, p. 012235.

[4] Iftikhar Ahmad et al. "Fake news detection using machine learning ensemble methods". In: Complexity 2020 (2020).

[5] Prabha, T. & Aisuwariya, T. & Kiran, M. & Vasudevan, Shriram. (2020). An Innovative and Implementable Approach for Online Fake News Detection Through MachineLearning. Journal of Computational and Theoretical Nano Science. 17.130-135. 10.1166/jctn.2020.8639

[6] Anushaya Prabha, T.; Aisuwariya, T.; Vamsee Krishna Kiran, M.; Vasudevan, Shriram K Journal of Computational and Theoretical Nanoscience, Volume 17, Number 1, January 2020, pp. 130-135(6).

[7] Monther Aldwairi and Ali Alwahedi. "Detecting fake news in social media networks". In: Procedia Computer Science 141 (2018), pp. 215–222.

[8] Marco L Della Vedova et al. "Automatic online false news detection combining content and social signals". In: 2018 22nd Conference of Open Innovations Association (FRUCT). IEEE. 2018, pp. 272–279.

[9] Pérez-Rosas, Verónica & Kleinberg, Bennett & Lefevre, Alexandra & Mihalcea, Rada. (2017). Automatic Detection of Fake News.

[10] Hadeer Ahmed1 & Issa Traorel, and Sherif Saad. "Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques". International Conference on Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments (2017)

[11] Gahirwal Manishaet.al; International Journal of Advance Research, Ideas, and Innovations in Technology ISSN: 2454-132X Impact factor: 4.295

[12] Traore, Issa & Saad, Sherif. (2017). Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. 127-138. 10.1007/978-3-319-69155-8_9.

[13] Mykhailo Granik, Volodymyr Mesyura, "Fake News Detection Using Naive Bayes Classifier", 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON).

[14] Horne, B.D., Adali, S.: This just in: false news packs a lot in the title, uses simpler, repetitive content in a text body, more similar to satire than real news. In: The 2nd International Workshop on News and Public Opinion at ICWSM (2017)

[15] Wang, W.Y.: Liar, Liar Pants on fire: a new Benchmark dataset for false news detection. arXiv preprint (2017). arXiv:1705.00648

[16] S. Volkova, K. Shaffer, J.Y. Jang, N. Hodas, in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), vol. 2 (2017), vol. 2, pp. 647– 653

[17] S. Volkova, K. Shaffer, J.Y. Jang, N. Hodas, in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), vol. 2 (2017), vol. 2, pp. 647–653

[18] Cody Buntain and Jennifer Golbeck. "Automatically identifying false news in popular Twitter threads". In: 2017 IEEE International Conference on Smart Cloud (SmartCloud). IEEE. 2017, pp. 208–215.

[19] Mykhailo Granik and Volodymyr Mesyura. "false news detection using naive Bayes classifier". In: 2017 IEEE first Ukraine conference on electrical and computer engineering (UKRCON). IEEE. 2017, pp. 900–903.

[20] Cade Metz. "The bittersweet sweepstakes to build an AI that destroys false news". In: Wired.com (2016).

[21] Rubin., Victoria, L., et al.: false news or truth? Using satirical cues to detect potentially misleading news. In: Proceedings of NAACL-HLT (2016)

[22] Nadia K Conroy, Victoria L Rubin, and Yimin Chen. "Automatic deception detection: Methods for finding false news". In: Proceedings of the Association for Information Science and Technology 52.1 (2015), pp. 1–4.

[23] S. Chhabra, A. Aggarwal, F. Benevenuto, P. Kumaraguru, in Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (ACM, 2011), pp. 92–101

[24] Allcott, Hunt, and Matthew Gentzkow. 2017. "Social Media and Fake News in the 2016 Election." Journal of Economic Perspectives, 31 (2): 211-36.

[25] Grinberg, N., Joseph, K., Friedland, L., Swire-Thompson, B., & Lazer, D. (2019). Fake news on Twitter during the 2016 U.S. presidential election. Science, 363(6425), 374–378. https://doi.org/10.1126/science.aau2706.