Fake News Detection and Analysis

Bowen Zhang[1,a,*], Chen Dai[2], Ziqing Deng[3] ,and Zenghan Jiang[4]

[1]*Faculty of Engineering and Information Technology, University of Melbourne, Melbourne, 3010, Australia*
[2]*College of Informations Science and Technology, Shihezi University, Shihezi, 832003, China*
[3]*School of Economics and Business Administration, Chongqing University, Chongqing, 400044, China*
[4]*Department of Economics, Dana and David Dornsife College of Letters, Arts and Sciences, University of Southern California, Los Angeles, CA 90089, USA*
a. 2049027754@qq.com
**corresponding author*

Abstract: The popularized use of social media accelerates the spreading of fake news. The overwhelming amount of fake news was a severe social issue during the 2016 presidential election and the first outbreak of Coronavirus in 2020. As controlling the spread of fake news is not practically workable, the detection of fake news is significantly valuable to solve this issue. In this paper, we conduct experiments to discover the effect of contextualized embedding of news content on counterfeit news detection. We also explore the features of fake news through two aspects: clickbait and sentiment.

Keywords: fake news detection, deep learning, contextualized embedding

1. Introduction

With the rapid development of information technology, the Internet has become the first choice to keep social connections. Social media wins people's preference because of its low cost, easy access, and rapid dissemination of information [1]. However, as a newer means of communication, the Internet is a double-edged sword. On the one hand, social media features enable information to spread highly efficiently. But on the other hand, it has become the perfect breeding ground for fake news. With such a large amount of online communication, it's difficult to determine truth versus fake. It's widespread for phony info to appear frequently and spread like a virus on the Internet. People are stuck in the information marsh, bothered by so much fake news in their daily lives, but they have no opportunity to get themselves out. Fake news permeates people's lives so profoundly across aspects where they are affected or misled by misinformation. This includes politics, everyday choices, and social activities, especially after the US presidential election in 2016 [2]. During the 2016 Presidential election, social media sites, such as Twitter and Facebook, were flooded with "fake news" [3]. Since then, fake news has attracted wide attention in public and academic circles. Based on this phenomenon, extensive research has been carried out to figure out the production, spread pattern, and detection of fake news. However, because of the complexity of social media, there are still many difficulties and gaps in capturing and controlling fake news.

In this research, we intend to apply transformer-based encoder-decoder architecture deep learning models on fake news detection to explore the effect brought by considering the con-textualized embedding of sentences. Also, we analyze the properties of fake news to summarize its features.

2. Literature Review

For a long time, people have expected to communicate with machines directly. Natural Language Processing (NLP) is a technology that enables people to communicate with computers using their natural language. Many propitious works have been made thanks to the utilization of artificial neural networks (ANNs) [4]. With the development of machine learning, all kinds of neural networks have been applied to NLP tasks. Kalchbrenner et al. used convolutional neural networks (CNNs), and Yao et al. upgraded the model by using more flexible graph convolutional neural networks [5,6]. Since NLP is becoming widely adopted, it is vital for information analysis from social media. Monti et al. applied non-Euclidean (graph- and manifold-structured) data to learn the propagation pattern of fake news on Twitter [7]. Oshi-kawa et al. used NLP in automatic phony news detection to detect the production of fake news on the Internet and effectively control the spread of fake news [8]. We will use these existing NLP theories and previous research to make our phony news detection model.

Shishah provided a brand-new BERT approach with a joint learning framework that integrates relational features classification (RFC) and named entity recognition (NER) [9]. Shishah and his partners transformed BERT, entirely using all hidden states after encoding parts of features as their primary encoder. They identified fake news detection as a fine-grained binary-classification task. This joint framework is supposed to distinguish the relations between elements in a long text format and give appropriate weight to entities such as names of people, cities, or countries. They implemented their proposed approach into two real-world fake news datasets, Politifact and Primedia. Shishah's model distinguishes be-tween entities involved in a long text to help to decide whether certain news should be flagged as true or false (fake). This was the first attempt to combine RFC and NER in a BERT model to detect fake news. In this paper, a Bert-based deep learning model is used to detect and uncover hidden news contents, significantly improving analysis accuracy and detection credibility.

The Naive Bayes model is based on Bayes' theorem for data classification. It assumes that the features are independent, simplifying the previous algorithm and making the model more effective. Although this condition is rare, the model has proved surprisingly well in practice [10]. Later research focused on loosening this restriction for more accuracy [11]. Kibriya et al. used locally weighted learning to improve the transformed weight-normalized complement naive Bayes (TWCNB) model [12]. Kim et al. found that Naive Bayes per-formed poorly in automatic text classification, so they proposed a feature weighting approach to replace the general feature selection method [13]. Their work showed a significant improvement in model performance. Zhang and Gao introduced an additional feature to help the original Naive Bayes model get higher accuracy in junk mail detection [14]. To further improve the accuracy of Naive Bayes, Jiang et al. proposed a new approach called deep feature weighting (DFW), which applied conditional probability estimates in feature weight to adjust the formula [11]. Although the original Naive Bayes model is quite mature, it is widely used in text analysis, and there have been many recent improvements. Wang first incorporated Naive Bayes into spam detection and showed a good performance [15]. Chakraborty et al. used Naive Bayes to evaluate text sentiment about COVID-19 [16]. In this article, Naive Bayes will play an essential role in capturing features to detect fake news.

3. Dataset

In general, one of the critical challenges in machine learning research, particularly in detecting certain kinds of textual information, is the collection of sufficiently rich and reliable datasets to train and test the algorithms. Since the concept of "fake news" is rather vague and subtle, we selected a few highly cited and convincing datasets on Kaggle to ensure their reliability and quantity.

3.1. Fake News Dataset

This (Getting Real about Fake News Kaggle dataset: https://www.kaggle.com/code/ohseokkim/fake-news-easy-nlp-text-classification/data) is text and metadata from fake and biased news sources around the web, and the dataset con-sists of 59,990 news or tweets between March 31, 2015, through February 18, 2018. This dataset contains four data sources: "fake and real news dataset," "fake news," "getting real about fake news," and "source based on fake news classification." As a result, there are 35,273 fake news articles and 24,717 accurate news articles.

This data was from professional journals such as the Journal of Security and Privacy and Lecture Notes in Computer Science [17,18]. These datasets seemed the most promising for preprocessing, feature extraction, and model classification. The reason is that the other datasets lacked the sources from where the article/statement text was produced and published. Our dataset contains a rich corpus of real and fake news, both in terms of reference and new type, with columns including news/tweet title, text, subject, publication date, authority, and, most importantly, the type (real or fake).

3.2. News Clickbait Dataset

Publishers often use online content (News Clickbait Dataset: https://www.kaggle.com/datasets/vikassingh1996/news-clickbait-dataset/ code) catchy headlines for their articles in order to attract users to their websites. These head-lines, popularly known as clickbait, exploit a user's curiosity gap and lure them to click on links that generally link to spam. The dataset contains 52,891 news or tweets with two columns: the first includes headlines, and the second has numerical clickbait labels. As a result, there are 32,739 clickbait and 20,290 non-clickbait. We use these data to test our model to analyze the relationship between clickbait and fake news.

Table 1: Dataset Description.

Fields	Dataset type	Description
Title	Fake News	The title of the news article
Text	Fake News	The text of the news article
Subject	Fake News	The subject of the news
Date	Fake News	The date at which the news was posted
Source	Fake News	The site url and country where the news is from
Type	Fake News	Tagged as either "fake" or "truth"
Headline	News Clickbait	The headline contains title of the news
Label	News Clickbait	Tagged as either "0" or "1"

Note: This table describes each data set, fake news, and news clickbait. The number of articles for the fake news and news clickbait datasets was 59,990 and 52,891, respectively, and ranged between March 31, 2015, through February 18, 2018.

4. Methodology

4.1. Approach

To study the importance of contextualized sentence embedding in fake news detection, we first build two classical machine learning models (Naïve Bayes and Logistic Regression) as baseline models. We fine-tune two pre-trained deep learning models (BERT and T5) and compare their performance. This section will discuss the steps taken to build fake news classifiers.

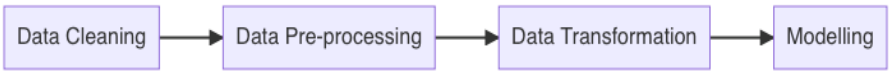

Figure 1: Methodology.

4.2. Preprocessing

From the data we collect, only titles and texts are extracted for training, and we only focus on meaningful news written in English. By significant, we mean the information should have a title, and its text should have more than one word. Further preprocessing steps vary among the models we adopt, but we split the data into 70%, 15%, and 15% for training, validating, and testing, respectively, for all purposes.

4.3. Naïve Bayes and Logistic Regression

The Naïve Bayes model is a supervised probabilistic model derived from Bayes' theorem (Formula 1). Given an instance T, which class c gives the highest probability (Formula 2). Since the denominator does not depend on c, we only need to choose c that maximizes the product in the numerator.

$$P(c|T) = \frac{P(T|c)P(c)}{P(T)} \quad (1)$$

$$\hat{c} = argmax_{c \in C} P(c|T) \quad (2)$$

There are three types of Naïve Bayes models: Gaussian NB requires continuous features and assumes Gaussian (normal) distribution; Bernoulli NB requires binary components (presence or absence of a word); Multinomial NB also takes the frequency of words into account and is suitable when we care about the number of times certain words appear [19]. In this research, we choose to use a Multinomial NB classifier.

The logistic regression model is also a supervised probabilistic model that translates a classification task into a regression task. The logistic regression model uses a logistic function to relieve the impact brought by the instances far from the decision boundary (outliers) and is denoted as Formula 3, where x is a vector of feature values and w is the corresponding feature weights. The training process is to iteratively update weights until the sum of cross-entropy errors per training instance is minimized. The instance is classified as "class 1" if the calculated probability is more significant than 0.5 and "class 0" otherwise.

$$P(Y = 1|x) = \frac{1}{1+exp(-x \cdot w)} \quad (3)$$

The Naïve Bayes model assumes all features are independent. Even though this assumption rarely holds, it still works well. In comparison, the Logistic Regression model does not make any

assumptions and usually leads to better performance. Both are proper models when it comes to binary classification tasks.

For these two models, we apply the unigram bag-of-words (BOW) model as input, where we concatenate the title and text, tokenize and lowercase the tokens, remove words that are not in the English dictionary, remove stopwords, and count the frequency of each remaining word. Then, we tune parameters (smoothing and regularization) on validation sets to find their optimal performances.

4.4. RoBERTa and GRU

Bidirectional Encoder Representations from Transformers (BERT) is an encoder-decoder model using the self-attention mechanism for NLP tasks developed by Google. It has two training objectives. First, a certain percentage of words are masked at random, and then predicted these masked words (MLM). Second, it learns relationships between sentences by predicting whether a sentence follows another. In this way, BERT captures deep bidirectional information and learns contextual representation. The pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for many tasks without substantial task-specific architecture modifications [20]. A Robustly Optimized BERT Pretraining Approach (RoBERTa) is a replication study of BERT pre-training with four modifications: training longer with larger batch size and more data, removing the next sentence prediction (NSP) objective, training on a longer sequence, and dynamically changing the masking pattern applied to the training data. RoBERTa is proven to match or exceed the performance of all the post-BERT methods [21].

Recurrent neural network (RNN) is an artificial neural network (ANN) type where embed-dings of arbitrarily sized inputs are allowed. The core idea of RNN is to process the input sequence one at a time by applying a recurrence formula. RNN is trained by backpropagation through time, and when the line gets longer, the backpropagated gradient vanishes. A Gated recurrent unit (GRU) is a gating mechanism in RNN and is introduced to solve this problem. It is like Long Short-term Memory (LSTM), but with fewer parameters [22].

In this approach, we break news text into sentences and join them with a SEP token (identify the ending of a sentence). After feeding into the RoBERTa model, we extract the embed-dings of the SEP tokens from the model's last hidden layer and iteratively put them through a GRU cell to get the final hidden state; then, we put this hidden state into a linear layer to perform classification. In this way, we purposely train our model to put sentence information into the SEP tokens, and by using GRU, we can get a contextualized embedding of the whole text.

4.5. T5

T5 model is another transformer-based encoder-decoder NLP model developed by Google. This model utilizes the idea of transfer learning and intends to convert all NLP problems into a unified text-to-text format. To achieve this goal, T5 is trained with maximum likelihood estimation (MLE) and cross-entropy loss, irrespective of the task. Unlike BERT, the objective of T5 is span prediction (mask spans) instead of MLM (mask words) and is trained on a larger dataset (750GB) that consists of "reasonably clean and natural English text" [23].

T5 has five different versions varying on parameter size. In this approach, we choose T5-large because of the limitation of GPU resources. To pre-process, we simply prepend texts with our research task, "detect:", and directly feed them into this pre-trained model.

5. Result

The result of our model is shown in Table 1. Here we focus on the F1 score of the "Fake" tag, as the focus on this research. For BERT and T5, we only run them for one epoch due to the time they take

(1 hour/epoch and 3 hours/epoch, respectively). It can be seen that all their scores are higher than 95%, and it's probably because the dataset we chose is too easy, or there might be some "indicators" showing true or fake (We give some interpretations about this in Appendix A.), but still, this contextualized-embedding method indeed is helpful, especially for T5, which correctly classifies all the test instances.

Table 2: Result.

Model	Naive Bayes	Logistic Regression	RoBERTa + GRU	T5
F1 Score	0.95	0.97	0.99	1
Precision	0.95	0.97	0.99	1
Recall	0.95	0.98	0.99	1

6. Feature Analysis

6.1. Clickbait

As social media is replacing other communication tools and becoming the most crucial method to propagate information, traditional barriers to publishing content (like a press to print newspapers or to broadcast time for radio or television) have disappeared. This has disappeared at least part of traditional quality control procedures [24]. Nowadays, a nearly limitless amount of information fills our online world, and everyone who has access to the Internet can post passages online without restriction. In this context, the media's authority to play gatekeeping is under threat [25, 26]. Under the pressure of competition, every medium must try to attract the readers' attention. At the same time, headlines are snapshots of the news. As many news outlets shift resources to digital forms of journalism, the function of news headlines has gained renewed importance: entice and engage new audiences [27]. This is so-called "clickbait".

People now live in an information "greenhouse", where information on social media can be manipulated easily. People can only see what they are expected to see. Clickbait is one of the methods to achieve this result, where people are led by clickbait, click on the news, and are subtly influenced by the content. There is no need to criticize the media for modifying headlines to make money from the clicks the readers make [28]. However, fake news will also be packaged to spread and amplify through the Internet, and the phenomenon is worthy of social attention. Considering this, our group did further research. We want to figure out the percentage of clickbait in real and fake news and find some patterns in the dissemination of phony information.

First, we use Naive Bayes to train the clickbait news dataset and acquire a clickbait discriminator. The accuracy of the model reaches up to 95.8%. Next, the discriminator is used to predict the real and fake news data sets, analyze the proportion of clickbait in the titles of real and fake news, and finally, draw a conclusion. In our result, 35.2% of fake news is de-signed as clickbait, which only accounts for 6% of accurate information. It is significantly lower than fake news. It indicates that editors will pay more attention to fake packaging news to attract public attention, which means the public is likelier to glance over fake news. This phenomenon deserves our vigilance.

6.2. Sentiment Analysis

The creators of fake news use a variety of stylistic tricks to promote the success of their creations, one of which is to stimulate the recipient's emotions, which leads to sentiment analysis [29]. Ajao et al.proposed a hypothesis based on empirical observations that there is a relationship between fake news or rumors and the sentiment of texts posted online [30]. So we decided to use the polarity and intensity of the opinion expressed in the text as a complementary element of the fake news detection method.

We used two of our collected datasets; one has fake news, and the other only has accurate information. Here are the results of processing the data using python and the VADER model. There are three types of storytelling values in the news industry — positive, negative, and silver-lining [31]. Our model will only output two types - positive or negative.

It is clear that the proportion of news with positive sentiment in both datasets is almost 54% (Table 2), and as far as we can see, there is no strong correlation between the news being real or fake and the sentiment it expresses. We believe that the expression of opinion is complex, and if we want to use emotion to determine actual or simulated news, we should need more characteristics. It is not enough to judge real or fake news by the criterion of positive or negative tendencies of news sentiment.

Table 3: Sentiment Result.

Dataset	Number of News	Number of Positive News	Number of Negative News
Fake	23481	12893(54.9%)	10588(45.0%)
Modified_True	21390	11601(54.2%)	9789(45.7%)

7. Conclusion

A dataset containing 59,990 news or tweets from 2015 to 2018 was formed. Naive Bayes and logistic Regression models concatenate text and title, make the case not sensitive, and re-move some distractions that would cause inaccuracy, like tokens and stop words. Texts were broken down into sentences and joined with distinguished indicators by building Bert and T5 deep-learning models. The feature analysis of the modified dataset resulted in a significant relationship between news with clickbait and fake news. And the sentiment analysis shows insufficient evidence to conclude that positive and negative tendencies are related to news authenticity.

Three future directions are set to enhance the value of this project. First, making news collected more time-sensitive means containing news from 2019 to now, especially for this cur-rent year. For example, the Russia and Ukraine war could be one of the topics. Secondly, making our news topic more comprehensive requires news from multiple datasets and covering more topics like economic news and sports news. The last direction is to ensure that our project is robust. To achieve this, run Bert and T5 multiple times and modify established models when processing new datasets from different sources and topics.

References

[1] *K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," ACM SIGKDD explorations newsletter, vol. 19, no. 1, pp. 22–36, 2017.*

[2] *E. Jardine, "Beware fake news," Governing Cyberspace during a Crisis in Trust essay. Waterloo, ON: CIGI. www. cigionline. org/articles/beware-fake-news, 2019.*

[3] P. N. Howard, G. Bolsover, B. Kollanyi, S. Bradshaw, and L.-M. Neudert, "Junk news and bots during the us election: What were michigan voters sharing over twitter," CompProp, OII, Data Memo, vol. 1, 2017.
[4] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 2, pp. 604–624, Feb. 2021.
[5] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," arXiv preprint arXiv:1404.2188, 2014.
[6] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in Proceedings of the AAAI conference on artificial intelligence, vol. 33, no. 01, 2019, pp. 7370–7377.
[7] F. Monti, F. Frasca, D. Eynard, D. Mannion, and M. M. Bronstein, "Fake news detection on social media using geometric deep learning," arXiv preprint arXiv:1902.06673, 2019.
[8] R. Oshikawa, J. Qian, and W. Y. Wang, "A survey on natural language processing for fake news detection," arXiv preprint arXiv:1811.00770, 2018.
[9] W. Shishah, "Fake news detection using bert model with joint learning," Arabian Journal for Science and Engineering, vol. 46, no. 9, pp. 9115–9127, 2021.
[10] I. Rish et al., "An empirical study of the naive bayes classifier," in IJCAI 2001 workshop on empirical methods in artificial intelligence, vol. 3, no. 22, 2001, pp. 41–46.
[11] L. Jiang, C. Li, S. Wang, and L. Zhang, "Deep feature weighting for naive bayes and its application to text classification," Engineering Applications of Artificial Intelligence, vol. 52, pp. 26–39, 2016.
[12] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited," in Australasian Joint Conference on Artificial Intelligence. Springer, 2004, pp. 488–499.
[13] S.-B. Kim, K.-S. Han, H.-C. Rim, and S. H. Myaeng, "Some effective techniques for naive bayes text classification," IEEE transactions on knowledge and data engineering, vol. 18, no. 11, pp. 1457–1466, 2006.
[14] W. Zhang and F. Gao, "An improvement to naive bayes for text classification," Procedia Engineering, vol. 15, pp. 2160–2164, 2011.
[15] A. H. Wang, "Don't follow me: Spam detection in twitter," in 2010 international conference on security and cryptography (SECRYPT). IEEE, 2010, pp. 1–10.
[16] K. Chakraborty, S. Bhatia, S. Bhattacharyya, J. Platos, R. Bag, and A. E. Hassanien, "Sentiment analysis of covid-19 tweets by deep learning classifiers—a study to show how popularity is affecting accuracy in social media," Applied Soft Computing, vol. 97, p. 106754, 2020.
[17] H. Ahmed, I. Traore, and S. Saad, "Detecting opinion spams and fake news using text classification," Security and Privacy, vol. 1, no. 1, p. e9, 2018.
[18] H. Ahmed, I. Traore, and S. Saad, "Detection of online fake news using n-gram analysis and machine learning techniques," in International conference on intelligent, secure, and dependable systems in distributed and cloud environments. Springer, 2017, pp. 127–138.
[19] G. Singh, B. Kumar, L. Gaur, and A. Tyagi, "Comparison between multinomial and bernoulli naïve bayes for text classification," in 2019 International Conference on Automation, Computational and Technology Management (ICACTM). IEEE, 2019, pp. 593–596.
[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.
[21] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," arXiv preprint arXiv:1907.11692, 2019.
[22] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv preprint arXiv:1412.3555, 2014.
[23] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu et al., "Exploring the limits of transfer learning with a unified text-to-text transformer." J. Mach. Learn. Res., vol. 21, no. 140, pp. 1–67, 2020.
[24] P. Bourgonje, J. Moreno Schneider, and G. Rehm, "From clickbait to fake news detection: An approach based on detecting the stance of headlines to articles," in Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism. Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 84–89. [Online]. Available: https://aclanthology.org/W17-4215
[25] S. Bowman and C. Willis, We media. How audiences are shaping the future of news and information, 66, 2003.
[26] D. Gillmor, We the media: Grassroots journalism by the people, for the people. Reilly Media, Inc, 2006.
[27] J. M. Scacco and A. Muddiman, "Investigating the influence of "clickbait" news headlines," Engaging News Project Report, 2016.
[28] A. Chakraborty, B. Paranjape, S. Kakarla, and N. Ganguly, "Stop clickbait: Detecting and preventing clickbaits in online news media," in 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). IEEE, Aug. 2016.

[29] *M. A. Alonso, D. Vilares, C. Gómez-Rodríguez, and J. Vilares, "Sentiment analysis for fake news detection," Electronics (Basel), vol. 10, no. 11, p. 1348, Jun. 2021.*
[30] *O. Ajao, D. Bhowmik, and S. Zargari, "Sentiment aware fake news detection on online social networks," in ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, May 2019.*
[31] *K. E. McIntyre and R. Gibson, "Positive news makes readers feel good: A "silver-lining" approach to negative news can attract audiences," South. Commun. J., vol. 81, no. 5, pp. 304–315, Oct. 2016.*

Appendix

Appendix A: Result

We understand that our fake news classifiers' results are uncommonly high, but we have thoroughly checked our code to ensure no test data is seen during training. Therefore, we did a statistical analysis of the dataset on a linguistic basis (count the frequencies of the top n most common non-stopwords of both tags to see the overlaps) to detect whether the cause of this phenomenon is due to their wording difference. Our analysis shows that 58% of the words are shared in the top 50 most common words and 61% in the top 100. Also, the top 10 most weighted words of Naive Bayes and Logistic Regression all appear in both tags' word lists, so there is no "oracle" telling the classifier what to choose.