An Approach to Detecting the Spread of False Information on the Internet Using Data Science Algorithms

L. Vitkova, K. Valieva, and D. Kozlov

1 St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences, 39 14th Liniya Street, Saint Petersburg 199178, Russian Federation
vitkova@comsec.spb.ru

2 The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, 22/1 Bolshevikov Avenue, St. Petersburg 193232, Russian Federation

Abstract. Today we are all witnessing the rapid immersion of society into the digital world. The amount of information is huge, and it is often difficult to distinguish normal news and comments from unreliable information. In this regard, the issue of detecting fake news and countering its spread becomes urgent. This task is not trivial for the following reasons: firstly, the volume of content that is created every day on the Internet is enormous; secondly, the detection system requires news plots that are obviously true; thirdly, the system must be able to analyze information in close to real time. The article presents a new approach to detecting the spread of false information on the Internet based on the use of data science algorithms. The concept of a fake news detection system includes 4 components and a data storage system. The article presents an experimental evaluation of methods implemented in the framework of the neural network training component and the detection of false information.

Keywords: False information detection · Fake news detection · Neural networks · Internet monitoring · Information dissemination · Internet security

1 Introduction

Today, we are all witnessing the rapid immersion of society into the digital world. Schools and universities switch to distance learning programs in case of urgent need. Organizations are implementing remote interaction services with customers and employees. The state and community are moving to online communication instead of offline meetings. At the same time, the number of papers devoted to detecting fake news is growing. For instance, on the Google Academy website [1], upon request “fake news detection” for 2016 we can see 4120 articles, for 2017 there are 5280 ones, for 2018 – 7250, and in 2020, 2379 papers have already been published only in 3 months (Fig. 1). Apparently, in 2020 we will see much more research than ever before devoted to the detection of fake news. It is worth paying attention to the fact that some of them, already in the 1st
quarter of 2020, are devoted to the problem of misinformation about the situation with COVID-19.

As an example, in [2] the authors raise a complex discussion about the fact that waves of panic and fears generated by fake news and misinformation on the Internet related to the spread of the virus lead to a deterioration of the physical and mental state of society.

In addition, in [3] the authors publish the results of a large-scale experiment. It shows that people do not think about the truth of the message and even spread it further on their social media pages. However, the authors propose the simple solution in [3], “notification that information is not verified” reduces the level of trust.

The issue of detecting and countering the dissemination of false information on the Internet requires a solution, especially given the appearance of various fakes during the COVID-19 coronavirus pandemic. It is necessary to develop and implement models and algorithms aimed at revealing false information on the Internet.

This article is a prolongation of the previous research [4, 5]. The authors propose an approach to detecting the spread of false information on the Internet using data science algorithms, which is described in more detail in Sects. 3 and 4. Initially, the writers raised the question: “Can data science algorithms successfully detect fake news on news aggregators, provided that the official position is published on official channels and then distributed to other sources?” The review of existing works and solutions (Sect. 2) was conducted with the experiments simultaneously; several datasets and models were considered.

![Fig. 1. Number of articles per year according to the Google Academy service. For 2020, the data are current for the 1st quarter of 2020.](image)

Therefore, in the beginning, experiments were realized with a dataset from an open repository [6]. An element of this dataset consisted of 2 news headlines and a fractional number from 0 to 5, which shows the semantic similarity of news headlines, the larger the number, the closer the texts are to each other. Convolutional neural networks were used to solve this problem, but the result was unsuccessful. The model was trained unstably and, in most cases, predicted a value of 2.5 regardless of the input data; the model’s prediction accuracy did not exceed 30%. It is assumed that the reasons for this behavior
are most likely due to the insufficient size of the source dataset for training the neural network.

Further analysis of other papers was carried out and the BERT model was taken as a basis; this model demonstrates a high level of detection accuracy, as shown in Sect. 4.

The approach proposed by the authors to detect fake news differs from existing ones in its taking into account official sources and information from them and comparing the news according to their short text parts broadcasted by news aggregators. This technique makes it possible to detect false information on time and counteract its dissemination.

2 Relevant Work Overview

Semantic analysis is a method aimed at building a semantic structure of a sentence consisting of semantic nodes and semantic relations. The purpose of the analysis is to construct these nodes, which are formed from the words of the original sentence. The basis for formulating hypotheses about the composition of semantic nodes is the information obtained as a result of syntactic analysis. The results of the analysis are presented as a semantic graph, the construction of which consists of a number of stages (initialization of semantic nodes and syntactic variants of fragments, construction of a set of dictionary interpretations of nodes, construction of time groups, construction of nodes in quotation marks, and so on).

Semantic analysis can be performed using various techniques, such as PROTAN and a wide range of other techniques. For example, semantic analysis is implemented in T-LAB Tools for Text Analysis, which is a computer technique that allows to perform three types of analysis: thematic analysis, comparative analysis, and adjacency analysis, as well as identify semantic patterns of words and the main ideas of the text. The process of working with text in this methodology includes text segmentation, keyword selection, and procedures designed to perform three types of analysis.

The paper [7] describes different types of fake news, which are clickbit, propaganda, opinion, humor, also news parody, forgery and photo manipulation. In addition to this approach, authors have intention to present several algorithms, which propose ways to spot fake news. They are quite simple, but effective:

- Evaluating sources of the news and other stories, checking the credibility of the source
- Consulting experts before sharing the information gathered in the Internet.

In [8] programs which were developed to perform repetitive tasks that gathered data (they were called bots) are described. In addition, in the information collected by bots researchers could see what sites people have been visiting, what they bought and how often purchases were made. Generally, it can be noticed that once bots were efficiently created the expansion of possibilities to collect and use gathered data happened. It is mentioned in [9] that people themselves are assisting in the spread of false information even though more than 50% of them might not see it. This is because of the feature of computer code, which, in some circumstances, could allow fake news to circulate and interfere people’s beliefs.
The article [10] includes an approach and an academic review on spreading false information in social media, more precisely, in Facebook. Authors pay attention on the process of circulating of fake news.

Studies describe the model that contains individuals who can be in one of the following states: ignorant, true-information spreader, fake-information spreader, and stifler. The research of [11] is directed to improve the accuracy of the existing techniques of detecting the spreading of false information.

Furthermore, [12] touches upon some machine learning techniques which can help to classify text documents. Fig. 2 shows the graphical representation of the text classification process.

![Fig. 2. Text classification process.](image)

The paper [12] describes different approaches of text classification based on machine learning techniques. They are:

- Naive Bayes
- Support vector machines (SVM)
- A fast decision tree construction algorithm [14], that was proposed by D. E. Johnson
- A method which improves performance of kNN
- TextCC.

Naive Bayes is often used in text classification because this method is very simple and effective [15]. However, the algorithm has disadvantages, it models text badly. The authors of [16] tried to solve this problem in their paper. They are suggesting that tree-like Bayesian networks could handle a text classification task in one hundred thousand variables with sufficient speed and accuracy.

Support vector machines (SVM) technique has its own advantages, but strong limitations. When the SVM method is applied to text classification, it provides excellent fidelity, but the recall is very weak. In order to improve the values of recall, the authors of [17] decided to describe an automatic process for adjusting the thresholds of generic SVM with better results.

Moreover, in [18] a method which improves performance of kNN by using well estimated parameters is presented. The main aim is to find out sufficient parameters.
Heui Lim resorted to propose and evaluate diversified the kNN approach with various decision functions, k values and feature sets to solve this issue.

TextCC is a training algorithm, which is submitted in [19]. In order to classify documents instantly the corner classification (CC) network is used. It is a kind of feed forward neural network.

Given the fact that the first experiment was unsuccessful, it was decided that the strategy of resolving the problem must change. The authors considered in [20] to take the BERT model as a basis for further explorations. BERT (Bidirectional Encoder Representations from Transformers) uses a “masked language model” (MLM) pre-training objective. That helps to mitigate the unidirectionality constraint, which is caused by two approaches (“feature-based” and “fine-tuning”) sharing the same objective function during pre-training, where they use unidirectional language models to learn general language representations.

In [21] the authors claim that BERT is the first representation model based on fine-tuning, which provides the most contemporary efficiency for a large set of sentence-level and token-level tasks, surpassing many architectures, focused on specific objectives.

BERT has been already used in many researches. For example, in [22] the authors performed systematic comparisons of different BERT+NMT architectures for standard supervised NMT. They also claim that the benefits of using pre-trained representations has been overlooked in previous studies and should be assessed beyond BLEU scores on in-domain datasets. Additionally, they compare different ways to train and reuse BERT for NMT.

As a result of the accomplished work [23] it was showed that BERT can be trained only with a masked LM task on the NMT source corpora and yield significant improvement over the baseline.

To summarize the information mentioned above it must be noticed that on the Internet nowadays there are many studies dedicated to confrontation to fake news, the popularity of this field is proceeding to increase. Nevertheless, the problem is that there are no such ready-made practical models and algorithms. It is required to develop the approach for detecting the spread of fake news in real time mode.

3 Proposed Approach

Direct text comparison is extremely resource-intensive and algorithmically complex. It eliminates the ability to quickly search for similar text. It is required to present a mechanism that allows to find similar text based on indexed fields in the database without lengthy and complex calculations.

The proposed model is supposed to contain methods of deep learning such as using the convolutional neural networks for text comparison. This is an approach to improve information security in the Internet. The main notion is to build a model that will determine matching one text to another, hereby the decision will be made: if news correspond to the more reliable source. The authors generate requirements for the matching and present them. If the inspected information does not meet the requirements, it can be defined as the false information.
A convolutional neural network is a special type of direct distribution neural network (Fig. 3). Direct propagation is understood as the propagation of signals through neurons from the first layer to the last. Authors observe in [24] that there can be quite many hidden layers in the network, depending on the amount of data and the complexity of the task.

![Fig. 3. A model of convolution neural network.](image)

The main feature of such networks is the presence of alternating layers of a type of “convolution layer – max pooling” and there can be a set of these layers. The convolution operation implies that each input fragment is incrementally multiplied by a small matrix of weights (the core) and the result is summarized. This amount is the output element, which is called the feature map. The paper [25] is showing that the weighted sum of inputs is passed through the activation function. Max pooling (the layer of subsampling) is a nonlinear compaction of the feature map with the passing a nonlinear transformation. The publication [26] deals with the fact that the pooling is interpreted as splitting the feature map into smaller matrices, the process of finding their maximum elements, so there is an increase in the “depth” of values.

The main concept proposed by the authors in this study lays in combining the data science and mechanisms information security by detecting the dissemination of false information on the Internet. A flowchart of the fake news detection system is offered in the Fig. 4.

- the monitoring component A connects to the RSS feeds of official news resources and collects all plots for the period Time = \(N\).
- the component B connects to the RSS feeds of other news resources and collects all plots for the period Time = \(N\).
- the storing component (BD) is entrusted with preservation and exchange of information between the components
- the component C is “The component of adaptation and retraining of the system”; it receives data from the component A through the database and submits it for training in the neural network
- the component D is “The component of detecting”; it receives data from the component B through the database and analyzes it.

Comparing news plots from official sources with others will make it possible to detect false information and make further decisions about how to counteract its dissemination. It is suggested that the solution proposed in [24, 25, 26] could be used within the flowchart.
from the Fig. 4. To detect fake news and false information, a neural network, which is constantly being re-trained in component C to analyze new news plots in component D, can be used.

4 Experiment

A news dataset on English language was used as a basis for the experiment [27]. It has the following structure:
- News headline
- The main text of the news
- The area that the news belongs to. The bigger part of the dataset consists of political news
- The date that coincides the publication of the news.

In the beginning, the dataset contained two files: true news and fake ones. In the data-preprocessing phase, they were combined into one structure to make the quantities of true news and fake news equal. With this dataset, which is evenly distributed across the classes, there is a possibility to use accuracy as a metric to evaluate the quality of the model. Moreover, another one column was added to the structure of the dataset. It maintained information about which class the news belonged to (fake news or true ones). The total size of the dataset constituted 34 267 cases. At first, the dataset was divided to 2 components: training one and testing samples in the ratio of 75% and 25% in accordance. BERT was chosen as the initial neural network model, which is a well-configured pre-trained neural network provided by Google. With this technique, the model does not need a large dataset to identify patterns; it is enough just to configure it to solve the current problem. By default, BERT returns a vector with a fixed dimension of 768 values for any size of input data, which contains complete information about the input data. In the present case, 3 linear fully connected layers with dimensions 256, 64, and 2 were added after BERT, respectively.

When the model was training, the PyTorch library [28] was used, which had already contained the BERT model and had provided convenient functionality for working with it. Before a text can be submitted to the model, it must be tokenized; it can be done by using the built-in BertTokenizer module. A diagram of the text preprocessing and model training is shown in Fig. 5. During this experiment only the text of the news itself is used to classify the news. This field has different lengths, so for convenience, before tokenization
each such text is separated into several fragments, the size of every one is 300 tokens. Then, in a period of the training, each of these fragments is successively transmitted to the input of the neural network. Learning is controlled by gradient stochastic descent, and the average square error is used as the loss function. Furthermore, it is recommended to use the capabilities of the video card for BERT training whenever it is possible, since this model is a deep neural network, which can take a considerable time to train. Based on the results of evaluating the model on a test sample, the accuracy of the prediction was 95%. Whereas the dataset under study has a uniform distribution across the classes, the precision and recall values were also 95%.

Figure 6 demonstrates a reduction in the level of losses during model training. It shows the absolute value of the loss function on the ordinate axis, and the iteration sequence number on the abscissa axis.

For all experiments, a laptop with the following characteristics QuadCore Intel Core i7-8565U, 4500 MHz (45 x 100); 16 GB DDR4-2666 DDR4 SDRAM was used. A powerful video card is required to speed up the model. On the current hardware, the duration of training the model and analyzing the test dataset took about 13 h.

![Diagram](image)

**Fig. 5.** The text preprocessing and model training.
5 Conclusion and Discussion

The article presents a new approach to detecting the spread of false information on the Internet using data science algorithms.

The developed concept of the detection system, proposed by the authors in this study, is to combine data science and information security mechanisms by detecting the spread of false information on the Internet. The discovery system concept includes 4 components and a data storage system.

Evaluation of methods implemented in the framework of “the component of adaptation and retraining of the system” and “the component of detecting”, performed during the experimental study, demonstrated the effectiveness of using the BERT model by means of the PyTorch library in this task.

Some areas of further research are the development, integration, validation and evaluation of all components of the developed concept. Testing the model on a much larger dataset and in a computing environment with a large number of computing resources.

It is assumed that the proposed approach can be integrated into not only systems for monitoring and countering the spread of false information, but also be part of antivirus systems, browsers, or parental control systems.

There is still a discussion about the possibility of components working in near-real-time mode, due to the huge amount of data. For example, an important part of the problem of spreading fake news is its appearance in social networks. However, the dynamics of the appearance of new posts and messages in these networks develops in such ways that the components of monitoring, training the neural network to recognize a new plot and detecting fake news can’t be triggered in a timely manner. The authors will demonstrate the results of testing the increased load on this model in future studies.

Acknowledgments. The work is performed by the grant of RSF #18-11-00302 in SPIIRAS.

References

1. Google Academy (2020). https://scholar.google.ru. Accessed 21 Apr 2020
2. Tasnim, S., Hossain, M.M., Mazumder, H.: Impact of rumors and misinformation on COVID-19 in social media. J. Prev. Med. Public Health 53(3), 171–174 (2020). https://doi.org/10.3961/jpmph.20.094

3. Pennycook, G., McPhetres, J., Zhang, Y., et al.: Fighting COVID-19 misinformation on social media: experimental evidence for a scalable accuracy-nudge intervention. J. Psychol. Sci. 31(7), 770–780 (2020). https://doi.org/10.1177/0956797620939054

4. Vitkova, L., Saenko, I., Tushkanova, O.: An approach to creating an intelligent system for detecting and countering inappropriate information on the Internet. Stud. Comput. Intell. 868, 244–254 (2019). https://doi.org/10.1007/978-3-030-32258-8_29

5. Vitkova, L., Doynikova, E., Kotenko, I.: Model of responses against unwanted, questionable and malicious information on the Internet. In: Abstracts of the VIII International Scientific-Technical and Scientific-Methodological Conference “Actual Problems of Information Telecommunications in Science and Education”, 27–28 February 2019, pp. 223–227 (2019)

6. Brmson: Semantic text similarity dataset hub (2020). https://github.com/brmson/dataset-sts. Accessed 24 Apr 2020

7. Sadiku, M.N., Eze, T.P., Musa, S.M.: Fake news and misinformation. Int. J. Adv. Sci. Res. Eng. 4(5), 187–190 (2018). https://doi.org/10.7324/IIASRE.2018.32728

8. Burkhardt, J.M.: How fake news spreads. Library technology reports “Combating fake news in the digital age”, vol. 53, no. 8, pp. 10–13 (2017)

9. Stavrou, V., Kandias, M., Karoula, G., et al.: Business process modeling for insider threat monitoring and handling. In: Lecture Notes in Computer Science, vol. 8647, pp. 119–131 (2014). https://doi.org/10.1007/978-3-319-09770-1_11

10. Yang, D., Chow, T.W.S., Zhong, L., et al.: True and fake information spreading over the Facebook. J. Phys. A: Stat. Mech. Appl. 505, 984–994 (2018). https://doi.org/10.1016/j.jphyasa.2018.04.026

11. Nguyen, N., Reiher, P., Kuenning, G.H.: Detecting insider threats by monitoring system call activity. In: IEEE Systems, Man and Cybernetics Society Information Assurance Workshop, pp. 45–52. Institute of Electrical and Electronics Engineers Inc. (2003). https://doi.org/10.1109/SMCSIA.2003.1232400

12. Ikonomakis, M., Kotsiantis, S., Tampakas, V.: Text classification using machine learning techniques. J WSEAS Trans. Comput. 4(8), 966–974 (2005). https://doi.org/10.11499/sicej1962.38.456

13. Kim, S.B., Rim, H.C., Yook, D.S., et al.: Effective methods for improving naive Bayes text classifiers. In: Lecture Notes in Computer Science, vol. 2417, pp. 414–423 (2002). https://doi.org/10.1007/3-540-45683-x_45

14. Lim, H.S.: Improving kNN based text classification with well estimated parameters. In: Lecture Notes in Computer Science, vol. 3316, pp. 516–523 (2004). https://doi.org/10.1007/978-3-540-30499-9_79

15. Kłopotek, M.A., Woch, M.: Very large Bayesian networks in text classification. In: Lecture Notes in Computer Science, vol. 2657, pp. 397–406 (2003). https://doi.org/10.1007/3-540-44860-8_4115

16. Shanahan, J.G., Roma, N.: Improving SVM text classification performance through threshold adjustment. In: Lecture Notes in Artificial Intelligence, vol. 2837, pp. 361–372 (2003). https://doi.org/10.1007/978-3-540-39857-8_33

17. Johnson, D.E., Oles, F.J., Zhang, T., et al.: A decision-tree-based symbolic rule induction system for text categorization. IBM Syst. J. 41(3), 428–437 (2002). https://doi.org/10.1147/sj.413.0428

18. Zhang, Z., Zhang, S., Chen, E., et al.: TextCC: new feed forward neural network for classifying documents instantly. In: Lecture Notes in Computer Science, vol. 3497, pp. 232–237 (2005). https://doi.org/10.1007/11427445_37
19. Vorobev, N.V., Puchkov, E.V.: Classification of texts using convolutional neural networks. Young Explor. Don 6(9), 2–7 (2017)
20. Devlin, J., Chang, M.W., Lee, K., et al.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies – Proceedings of the Conference, vol. 1, pp. 4171–4186 (2019)
21. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: 31st International Conference on Machine learning, vol. 4, pp. 2931–2939 (2014)
22. Clinchant, S., Jung, K.W., Nikoulina, V.: On the use of BERT for neural machine translation, pp. 108–117. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/d19-5611
23. Lample G, Conneau A.: Cross-lingual language model pretraining. arXiv preprint arXiv:1901.07291 (2019)
24. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. 9(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735
25. Bigelow, J.L., Edwards, A., Edwards, L.: Detecting cyberbullying using latent semantic indexing. In: Proceedings of the 1st International Workshop on Computational Methods for Cyber-Safety, pp. 11–14. Association for Computing Machinery (2016). https://doi.org/10.1145/3002137.3002144
26. Bouvrie J.: Notes on convolutional neural networks (2016). https://cogprints.org/5869/index.html. Accessed 21 Apr 2020
27. Ahmed, H., Traore, I., Saad, S.: Fake and real news dataset (2018). https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset. Accessed 26 Apr 2020
28. PyTorch (2020) https://pytorch.org. Accessed 26 Apr 2020