Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 192 (2021) 2893–2902

Procedia Computer Science

www.elsevier.com/locate/procedia

25th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Rapid detection of fake news based on machine learning methods

Barbara Probierz[a,*], Piotr Stefański[a], Jan Kozak[a]

[a]*Department of Machine Learning, University of Economics in Katowice, 1 Maja, 40-287 Katowice, Poland.*

Abstract

Nowadays, it is very important to quickly recognize the false information referred to as fake news. This is especially important in the case of news appearing on the Internet because of its wide and rapid spreading. It is equally important to be able to initially classify news as fake or true based on the title itself. In this paper, we propose an approach to classifying news based on the title without analyzing the other aspects. The obtained results will be compared with classification based on the whole news text. The goal of this work is to propose a method that balances between data analysis time and quality of classification in fake news prediction. We use natural language processing (NLP) to describe the title and text of the news. This is a complex process, requiring good analysis to be applied to classification. Therefore, the use of complex classifiers – in this case, classical ensemble methods – has been proposed in order to achieve a high quality of classification (measured by popular measure). In this paper, we present analyses of a real data set and results of news classification using the proposed model – including an ensemble of classifiers and single classifiers.

Keywords: ensemble methods; fake news; machine learning; natural language processing.

1. Introduction

People have always tried to communicate with each other. Initially, it was simple information passed on to each other verbally, but with the development of technology, more recent methods of disseminating information emerged. Unfortunately, not all information provided was true. Often false information was spread as a kind of gossip to confuse the enemy or harm the opponent. In particular, a lot of false information was created for propaganda purposes for political reasons or for financial gain.

Similarly, today, in the Internet age, we are inundated with a huge amount of news about world events in the form of articles made available in the media, and we often cannot tell whether the information presented is true or false. For this reason, a branch of research has emerged that deals with evaluating the veracity of information and detecting false information called fake news [11].

* Corresponding author.
E-mail address: barbara.probierz@ue.katowice.pl

10.1016/j.procs.2021.09.060

Fake news is false or partially false news [11], often of a sensational nature, created for political, financial and ideological reasons, in order to mislead the recipient or to draw attention and gain publicity [30]. Fake news, usually in the form of articles, is published in traditional media, but recently more and more fake news can be found in social media, making it spread at lightning speed and on a large scale [28]. In addition, in order to attract the attention of recipients, a common technique used to spread fake news is to include catchy headlines, the so-called clickbait [4, 33].

Based on the analysis of the presented problem, the goal of this paper is to develop a new model for the quick discovery of fake news based on the news title without the need to analyze the whole content of the article. Such an approach is particularly important in the case of news prediction in social media due to the instant and wide spread of different information.

Moreover, we hypothesize that the analysis of the real-world data sets and the application of the proposed model based on natural language processing (NLP) and machine learning algorithms for classification allow quickly identifying fake news by its title alone. While additional text analysis further improves the prediction of true or fake news. The proposed method is based on NLP methodologies and machine learning techniques, both single classifiers and ensemble methods.

To verify the proposed model was used real data set "ISOT Fake News Dataset", which is made available by the University of Victoria, Canada [1]. The real news comes from Reuters.com, while the fake news comes mostly from sites marked by Politifact.com as untrustworthy sources.

Our contribution is to show that the detection of fake news can be effective on the basis of title itself, without the need to analyze the entire text, which significantly reduces the time of decision-making, especially when using machine learning algorithms. Such a solution is important for the widespread dissemination of information that harms society. Additionally, our method of transforming natural language text into a matrix of token counts is used to apply popular machine learning classifiers based on decision tables. This is the preliminary research related to possibility of building a multi-criteria model of fake news detection.

The remainder of this paper is organized as follows. Section 1 comprises an introduction to the subject of this article. Section 2 provides an overview of related works on the definition and detection of fake news. In section 3 we present natural language processing methods. Section 4 describes machine learning algorithms, in particular ensemble methods. In Section 5 we explain our methodology for detection fake news. Section 6, we present and discuss experimental results. Finally, in Section 7 we conclude with general remarks on this work and a few directions for future research are pointed out.

2. Fake news and related works

The subject of fake news is very extensive and covers many scientific fields. The research on detecting fake news is relatively new field. For these reasons, no single definition of fake news, accepted in all areas, has yet been developed. This phenomenon is characterized by a large semantic capacity and often occurs under different names. Some researchers and experts prefer the word "disinformation" over the term fake news. Facebook, the largest social network in the world, calls this content false news, and Snopes, one of the oldest English-language fact-checking websites, prefers to replace it with junk news.

By definition, fake news is usually false or partially false information disseminated on the Internet and in traditional media with the aim of misleading the recipient. However, by analyzing individual names and their meanings, it is possible to identify three main groups of false news that were defined in the work [25]. The first group of fake news is completely fabricated content in which no fragment is real. Veronica Perez-Rosas et al. focused their work [23] on developing a mechanism to automatically identify false files in online news. They developed classification models that were based on lexical, syntactic combinations, and semantic information, as well as features representing text readability properties. In addition, the authors performed a comparative analysis of automatic and manual identification of fake news, where the model results achieved comparable accuracy to the human ability to detect fake news.

The second group of fake news is satirical information, which is a joke or a parody. Rubin et al. [26] proposed an SVM-based algorithm with 5 predictive traits (absurdity, humor, grammar, negative affect, and punctuation) to classify the satirical data set. Rashkin et al. [24] compared the language of real news with that of satire, hoaxes, and propaganda to find linguistic characteristics of the untrustworthy text.

On the other hand, the third group consists of real news that have been taken out of context or have been imprecisely described. Unfortunately, large-scale hoax attacks are creative, unique, and often cross-platform, which may require methods that go beyond text analysis. For such information, a prototype of an innovative technology proving the origin of the captured digital media was created [17]. The authors believe that its application and the use of blockchain technology and standardized metadata introduces an innovative approach to overcoming the falsehood in reporting and the origin of the media resources used in it.

Many people do not distinguish fake news from real news [21]. However, the impact of this kind of news on the audience has not yet been sufficiently researched [20] and knowledge of its political, social, economic, and financial implications is still insufficient. Most studies concern false content posted on social networking sites [2]. First of all, there is information posted on Twitter, which results, among others, because of the openness of its application programming interface (API) and its popularity among politicians and journalists [13]. The second-largest fake news outlet is Facebook, where the authors of [14] investigated the individual characteristics of sharing fake news during the 2016 US presidential campaign.

Research communities expressed concern about the flood of misinformation and introduced automated fake news identification solutions based on natural language processing techniques (NLP). These models look simply at linguistic characteristics such as grammar features, word patterns, term count, and appearance of certain expressions. Granik and Mesyura [12] presented a simple approach for fake news detection using naive Bayes classifier and achieved decent results considering the relative simplicity of their model. Horne and Adali [16] analyzed difference of fake and real news in title features, complexity, and style of text. Elaboration Likelihood Model was considered as a theory to explain the spread and persuasion of fake news.

In addition to analyzing the entire content of the information, many researchers have focused on analyzing headlines, the so-called clickbait. Chen et al. [8] examined potential methods for the automatic detection of clickbait. Methods for recognizing both textual clickbaiting cues and nontextual ones including image and user behavior were surveyed. Bourgonje et al. [4] presented a system for detecting the stance of headlines with regard to their corresponding article bodies. The approach could be applied in fake news, especially clickbait detection scenarios.

3. Natural language processing

Natural language processing is a field of science that studies the ability to use a computer to understand and manipulate natural language text or speech that would help a person. Engineers working on this problem aim to gather knowledge about how a person uses language and how to perceive and understand it. Based on this knowledge, one is able to develop appropriate tools and techniques to create computer systems that could understand and use natural language. The basics of natural language processing cover areas outside of science, namely knowledge of linguistics or psychology plays a great role, so the competence and knowledge of teams working on NLP-based systems is very extensive [9].

The process of preparing data for analysis is always the longest stage in working on a machine learning model. Experts agree that this step can take up to 80% of the time [18]. Therefore, in our work, methods of preparing the text for analysis were used, which are aimed at standardizing the text, cleaning it, and to some extent, generalizing it in order to reduce variability.

3.1. Tokenization

The $n - gram$ model is a language model for identifying and analyzing features used in language modeling and natural language processing [31]. By $n - gram$ we define a contiguous sequence of elements of length n. It can be a sequence of words, bytes, syllables or characters. The most commonly used $n - grams$ models for text categorization are $n - grams$ based on words and characters [32]. In this work, we use $n - grams$ based on words to represent document context and generate document classification functions.

3.2. Stop words

Stop words is a list of words that are filtered out before or after the natural language (text) data is processed [19]. While stop words typically refer to the most common words in a language, there is no one-size-fits-all stop words list used by all natural language processing tools, and not even all tools use one. Some tools specifically avoid removing these ignored words in order to support the phrase search.

3.3. Stemming and lemmatization

In order to limit the variability of the set consisting of tokens, which represents the processed text, two methods of token normalization are used. The first method is stemming, which is used to extract the prefix and suffix from words and then replace similar words with the same in the base form [3].

The second method of token normalization is the lemmatization process, which aims to reduce the word to its basic form. The result of lemmatization is a vector containing only unchanged forms of words. Several algorithms can be used in the lemmatization process, one of them uses the corpus of the processed language, which includes pairs of inflected words and their basic forms [29].

3.4. Word weighting measures

During the analysis of text data, there are instances of words that appear in many analyzed documents with high frequency. Such words usually do not carry valuable or distinctive information. Additionally, too much redundancy of terms, words, or longer phrases leads to a high level of computational burden on the learning process [15]. Moreover, irrelevant and redundant characteristics can negatively affect the accuracy and performance of classifiers. Therefore, in this work, we examined two different feature selection methods, namely term frequency (TF) and inverse term frequency (TF-IDF) document frequency.

Term frequency (TF)

Term Frequency (TF) is an approach that uses the number of words appearing in documents to determine the degree of similarity between the documents [22]. For this purpose, the process of weighting terms is carried out, i.e. determining the weights as the degree of belonging to the document, taking into account the frequency of the term in the text. In this way, each document is represented by groups of words. Term Frequency is calculated using the formula:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{1}$$

where $n_{i,j}$ is the raw count of a term t_i in the document d_j and the denominator is the sum of the raw count of all $n_{k,j}$ terms in the document d_j.

Term Frequency-Inverted Document Frequency (TF-IDF)

In order to reduce the weights of less significant words in a set of words, a technique called Term Frequency - Inverse Document Frequency (TF-IDF) is used. The TF-IDF measure informs about the frequency of terms occurrence, taking into account the appropriate balance of the meaning of the local term and its meaning in the context of the complete collection of documents [22].

The Inverse Document Frequency (IDF) in the text is the ratio of the number of processed documents n_d to the number of documents containing at least one occurrence of the term $\{d : t_i \in d\}$ and is expressed by the formula:

$$idf_i = \log \frac{n_d}{\{d : t_i \in d\}} \tag{2}$$

The value of TF-IDF is calculated as the product of the frequency of terms $tf_{i,j}$ with the inverse frequency in the text idf_i, which is expressed by the formula:

$$(tf\text{-}idf)_{i,j} = tf_{i,j} \times idf_i \tag{3}$$

By means of the TF-IDF algorithm, the weights of frequently occurring words in most documents are significantly lowered and become lower than words which, for example, appear several times but in one document [22].

4. Classifiers learning and ensemble learning

There are many machine learning algorithms that are well known for their high performance in classification tasks. Typically, this involves determining which of the two classes to assign the input data set. Some machine learning techniques such as SVM are used to analyze continuous data and define patterns in order to classify texts. In addition, the CART algorithm used to construct decision trees can be distinguished due to the way of data representation.

Another group of learning algorithms is the ensemble methods proposed by Breiman [5, 6], which construct a set of many individual classifiers and combine them to classify new data taking into account the weighted or unweighted voice of their predictions. Classifier assemblies are often much more accurate than the individual classifiers that compose them. Ensemble methods work by repeatedly running the base algorithm and formulating a vote based on the resulting hypotheses. Popular representatives of aggregation methods for groups of classifiers are bagging, boosting and random forests algorithms.

Support Vector Machine (SVM) is a machine learning technique that allows analyzing data and determine patterns in order to classify, which involves determining which of the two classes to assign a set of input data to. For the process of learning support vector machine, a training set is required, in which each element of the set has an indication of which class it belongs to. The obtained SVM model represents the data from the training set, separated from each other by a border with the widest possible margin, i.e. the distance from this hyperplane. Support vector machine was first introduced by Vladimir Vapnik in [10].

Classification and regression trees (CART) is an algorithm used to construct decision trees. This algorithm was first proposed by Breiman et al. in 1984 [7]. For the CART algorithm, Breiman et al. proposed two criteria for the splits: Gini and the twoing criterion.

The splitting criterion goal is to find the best test that will divide the data analyzed in the node into two, maximally homogeneous (in terms of the decision class) parts. This is by far the most difficult and complex stage in constructing decision trees. The exact interpretation of both criteria and the statistical justification for such a solution was given by Breiman et al. in [7].

The boosting method consists of creating a group of classifiers whose individual accuracy is not satisfactory, linked together to obtain a model with better characteristics. This approach was first proposed by Schapire in 1990 [27]. In particular, this approach involves building a good learning set based on weak subsets of learning. In this phase, resampling is strategically adjusted to provide the most informative training data for each successive classifier.

The bagging approach, firstly introduced by Breiman [5], is based on bootstrap aggregation and on multiple construction of classifiers on the basis of data subsets (bootstrap samples) generated from the whole learning set. Assuming that the results of the base classifiers are independent of each other, each of the base classifiers gives exactly one vote and ultimately simple voting determines the classification of the samples. For each instance, the class chosen by most classifiers is the ensemble decision.

The random forests can be seen as an improvement of bagging developed by Breiman [6], based on comparisons of bagging and boosting. It should also be noted that, unlike the two previously discussed ensemble methods, in random forests, decision trees must be used as single classifiers. The first step in constructing random forests is to create a training subset from which decision trees will be constructed (for each individual). A decision tree is then constructed based on each training subset. Unlike standard tree construction methods, in random forests, during test selection for each node (separately), there is a random draw of attributes that will be taken into account when making the split. Therefore, each split is made based on a different set of attributes.

5. Research methodology

A new model has been developed to achieve the goal of predicting new news. It is related to studying the problem, the resulting data processing, and proposing a method of learning the algorithm. This is possible through problem analysis, as described in sections 1 and 2. And also an in-depth data analysis, the most interesting of which are described in the section 6.2. The proposed solution is presented as Algorithm 1.

In the first step of the solution, data preprocessing is performed. From the data described in the section 6.1, the content is retrieved and initially cleaned – it includes making all characters the same case, and removing digits and non-alphabetic characters.

Algorithm 1: Proposed approach

Input: *content* – list of news titles or texts
Input: *selected_attribute* – content to learn (title or text)
Input: *selected_algorithm* – selected machine learning algorithm
Output: *classifier*

```
if selected_attribute == 'TITLE' then
    content := select_text(content, 'TITLE');
else
    content := select_text(content, 'TEXT');
content := clear_text(content);
content := tokenize_text(content);
data_set := vectorize_text(content);
train_set, test_set = split_data(data_set);
classifier := model_learning(train_set, selected_algorithm);
model_testing(test_set);
result classifier;
```

Then tokenize text is performed, including the removal of 326 stop words, such as 'per', 'well', 'whereas',' yet ',' 's',' every ',' as', 'of', 'is' etc. As a result of the performed actions, it is possible to convert a set of text documents into a matrix of token counts - saved as data set.

The second step is to apply the prepared matrix of token counts (data set) to learn algorithms. For this purpose, the mixed data (true and fake news) is divided into a training set and a test set. Depending on the selected variant (learning on titles or on text – differences in results are described in the section 6) and the selected algorithm, the next steps of machine learning follow. The selected algorithm is trained on training data, and the resulting classifier is used to predict new data (a test set is used to determine its quality of classification).

The detailed solutions used in the real model are described in section 6.

6. Results of the computational experiments

We present several observations resulting from the experiments that were made to test the proposed model and allow to verify the research hypothesis. So we check whether the use of the classification algorithms (including ensemble methods) in the prediction of true or false news allows you to identify the fake news after the title. Additionally we check, that content analysis improves news prediction even more, as true or false.

To compare the quality of classifications, experiments with both ensemble methods and individual classifiers were carried out. Each prediction was assessed using several classic measures of quality of classification, which allows for a full assessment of the proposed solution.

6.1. Experimental design

The developed approach was tested on a real data set containing news, both political and world news. The original data set [1] contains 21417 real news cases and 23481 fake news cases described by the attributes: title, text, subject, date.

In our considerations, according to the model described in section 5 data set was limited to title or text only and in each case, decision (true or fake news). This solution allows checking whether the title analysis, without more information, allows for initial classification of the news. Then, it allows defining how the content of the news influences the final decision.

The experiments were carried out using the train and test method, with the set being divided into 70% of the training set and 30% of the test set. Such a solution allows both the assessment of the classification quality and the simulating of the real situation, where more data can be used for learning than for prediction.

The classic ensemble methods were selected for the analysis: random forest, boosting and bagging (described in section 4) and single classifiers: decision tree (in this case, the CART algorithm) and support vector machine (SVM).

Table 1. Confusion matrix

	Predicted positive	Predicted negative
Positive examples (P)	True positive (TP)	False negative (FN)
Negative examples (N)	False positive (FP)	True negative (TN)

The quality of the classification was presented on the basis of the following measures that can be determined on the basis of the confusion matrix (Tab. 1): accuracy (Eq. (4)), precision (Eq. (5)), recall (Eq. (6)) and f1–score (Eq. (7)).

It should be noted that the accuracy of the classification is a measure used to estimate what number of objects was properly classified. In the situation where the quality of classification of a single decision class is more important, precision and recall should be rather used. By using precision, it is possible to estimate, what is the confidence of assuming, that the object from a given decision class will be properly classified. While the recall is used to estimate, what number of objects from a given decision class was properly classified.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

$$precision = \frac{TP}{TP+FP} \quad (5)$$

$$recall = \frac{TP}{TP+FN} \quad (6)$$

$$f1 = \frac{TP+TP}{TP+TP+FP+FN} \quad (7)$$

All algorithms were implemented in Python programming language with popular libraries as: pandas, numpy (for reading and simple operations on data); re, string (for text transformation); nltk, spacy (for text stemming and removing stop words); matplotlib, prettytable (for data visualizations); sklearn (for vectorizing and splitting data, also for learning and evaluation algorithms). All operations were running on Google Colab, which may be used for free. Resources were sharing in the Colab tool but initially during execution, an environment containing 12 GB of RAM and Intel Xeon processor within two cores clocked at 2.30GHz, were assigned.

6.2. Data set analysis

One part of the problem posed in this paper is to determine whether the title alone allows for a good prediction of fake news. This is mainly due to the running time of the algorithm and the access (in a real situation) to the news text. It should be noted that the vectorizer transform time for the title itself takes an average of 15 seconds, while for text it is already over 420 seconds.

Therefore, a data analysis was carried out primarily in terms of news titles. In Fig. 6.2 is presented tokens existing only on true news with count weight – 6.2 (a) and only on fake news with count weight – 6.2 (b). As can be seen, the distribution of tokens in both groups remains in a similar ratio. In each case, one token occurs very frequently, and the others are of a similar level. This shows that distinctive tokens can be found in both fake and true news.

The analysis also identified crucial tokens existing in the fake news title based on weights: TF-IDF, TF, and binary. The results obtained for each measure look similar, and Fig. 6.2 shows the results for TF-IDF weight (true news – Fig. 6.2 (a) and fake news Fig. 6.2 (b)). Again, similar distributions of tokens can be seen, which should allow initial classification of the news based on titles alone.

6.3. Learning algorithms based on the news titles

The first of the analyzed approaches was related to title-based learning. This approach allows the classification of news without the need for lengthy analysis. The results related to this experiment are presented in Tab. 2 – the quality of classification and Tab. 3 – the time required to learn the algorithm.

The obtained results should be analyzed together, because these criteria (quality and time) may be of importance. The values of selected measures of quality of classification presented in Tab. 2 allow confirming that learning based on titles alone allows obtaining satisfactory results.

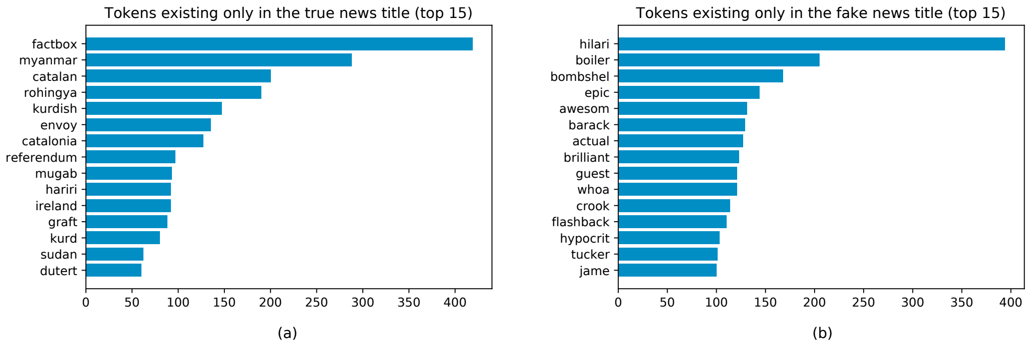

Fig. 1. Tokens existing only in title on: (a) true news with count weight (b) fake news with count weight

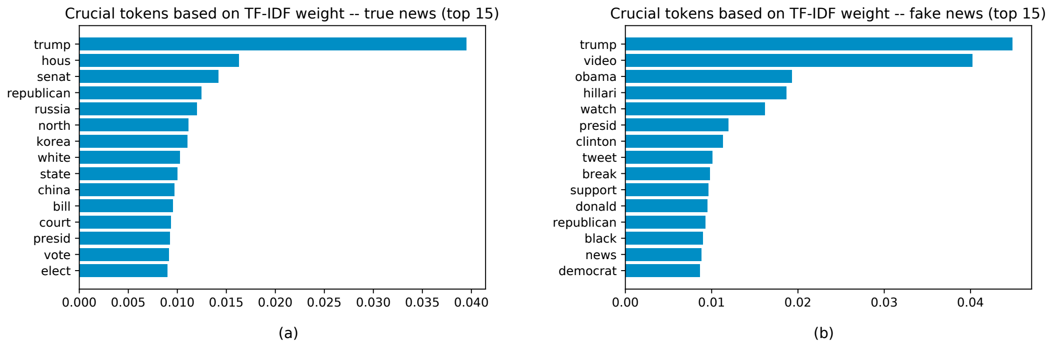

Fig. 2. Crucial tokens based on TF-IDF weight existing in: (a) the true news title (b) the fake news title

The best results for all measures used are obtained when the SVM algorithm is used (accuracy 0.9419). However, it should be noted (Tab. 3) that the running time of the SVM algorithm is the largest. It takes an average of 186.12 seconds, which is more than 78 times worse than the fastest of the algorithms – AdaBoost. It should be noted, however, that the values of quality of classification evaluation for AdaBoost are much lower (accuracy 0.8227).

The random forest algorithm, on the other hand, is a good balance between time and quality of classification. It allows classification with an accuracy of 0.9269 (while other measures reached about 0.91–0.95) with a running time of 7.82 seconds, which is more than 3 times slower than the AdaBoost algorithm, but almost 24 times faster than the SVM algorithm. Thus, it can be said that the classification accuracy worse by 1.5% is compensated by the significantly better performance of the ensemble methods – random forest algorithm compared to the single SVM classifier.

Table 2. Quality of the classification of algorithms learned based on the titles news (the best results in boldface)

	accuracy rate	precision real news	precision fake news	recall real news	recall fake news	f1–score real news	f1–score fake news
CART	0.8837	0.8758	0.8908	0.8780	0.8888	0.8769	0.8898
SVM	**0.9419**	**0.9302**	**0.9527**	**0.9479**	**0.9365**	**0.9390**	**0.9445**
Random forest	0.9269	0.9073	0.9456	0.9411	0.9141	0.9239	0.9296
AdaBoost	0.8227	0.7485	0.9306	0.9400	0.7179	0.8334	0.8105
Bagging	0.8984	0.8899	0.9060	0.8954	0.9011	0.8926	0.9035

6.4. Learning algorithms based on the news text

The next solution step was research related to text-based learning. This solution requires access to the entire news text and increases the analysis time. However, increasing the amount of information has an impact on the quality of

Table 3. Algorithm learning time in seconds – learning by title post

CART	SVM	Random forest	AdaBoost	Bagging
7.82	186.12	21.00	2.38	42.65

classification. The results related to this experiment are presented in Tab. 4 – the quality of classification and Tab. 5 – the time required to learn the algorithm.

In this case, the results of the quality of classification (Tab. 4) are no longer so clear-cut, with differences between the algorithms in each of the measures not exceeding 1%. When analyzing the results statistically, however, the Bagging algorithm performs best, followed by CART, AdaBoost, random forest, and the worst is SVM (which had the best results for the title alone).

These slight differences in the values of the quality of classification mean that the learning time of the algorithm is important (Tab. 5). In this case, the SVM algorithm also works the longest (on average, as long as 1384 seconds). The CART, AdaBoost, and random forest algorithms require a similar amount of time (about 30-40 seconds, which is up to 45 times less than the SVM). The bagging algorithm takes slightly longer – about 100 seconds.

Table 4. Quality of the classification of algorithms learned based on the news text (the best results in boldface)

	accuracy rate	precision real news	precision fake news	recall real news	recall fake news	f–score real news	f1–score fake news
CART	0.9954	**0.9963**	0.9946	0.9941	**0.9966**	0.9952	0.9956
SVM	0.9889	0.9862	0.9913	0.9905	0.9873	0.9884	0.9893
Random forest	0.9913	0.9887	0.9937	0.9932	0.9896	0.9909	0.9917
AdaBoost	0.9950	0.9926	**0.9973**	**0.9971**	0.9932	0.9948	0.9952
Bagging	**0.9964**	0.9960	0.9967	0.9964	0.9963	**0.9962**	**0.9965**

Table 5. Algorithm learning time in seconds – learning by text post

CART	SVM	Random forest	AdaBoost	Bagging
30.39	1384.10	40.66	37.78	109.67

7. Conclusions

The goal in this paper was to develop a new model to quickly discover fake news based on the news title alone without analyzing the entire news text. While for comparison purposes to the proposed method, whole news text analysis was also performed. NLP techniques were used to analyze the news title and text and eventually, machine learning algorithms, both single classifiers, and ensemble methods were used for prediction.

Observations resulting from the conducted experiments allowed confirming the hypothesis put forward by the authors. The application of the proposed model allowed us to obtain good results after the initial analysis of the news titles. However, the analysis of the text of articles allows for a very good classification of true or fake news. The problem is the running time of the SVM classifier, but the use of the random forest algorithm at the initial stage of experiments is a good trade-off between quality of classification and running time.

Future work may involve developing a multi-criteria model for a two-step analysis. It is possible to initially verify fake news based on the title and then perform a thorough analysis based on the text-only for selected news. Furthermore, the multi-criteria model can be based on the selection of appropriate features, i.e. quality of classification and running time to perform quick and efficient identification of true or fake news based on the title alone or slightly longer but very accurate analysis of articles based on the text. Additionally, it can be considered which of the considered classification algorithms could be used in an online learning scenario in which new fake or true news are being collected.

References

[1] Ahmed, H., Traore, I., Saad, S., 2018. Detecting opinion spams and fake news using text classification. Security and Privacy 1, e9.
[2] Allcott, H., Gentzkow, M., 2017. Social media and fake news in the 2016 election. Journal of economic perspectives 31, 211–36.
[3] Amirhosseini, M.H., Kazemian, H., 2019. Automating the process of identifying the preferred representational system in neuro linguistic programming using natural language processing. Cognitive processing 20, 175–193.
[4] Bourgonje, P., Schneider, J.M., Rehm, G., 2017. 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, pp. 84–89.
[5] Breiman, L., 1996. Bagging predictors. Machine Learning 24, 123–140.
[6] Breiman, L., 2001. Random forests. Mach. Learn. 45, 5–32.
[7] Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J., 1984. Classification and Regression Trees. Chapman & Hall, New York.
[8] Chen, Y., Conroy, N.J., Rubin, V.L., 2015. Misleading online content: recognizing clickbait as" false news", in: Proceedings of the 2015 ACM on workshop on multimodal deception detection, pp. 15–19.
[9] Chowdhary, K., 2020. Fundamentals of Artificial Intelligence. Springer.
[10] Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine Learning .
[11] Gelfert, A., 2018. Fake news: A definition. Informal Logic 38, 84–117.
[12] Granik, M., Mesyura, V., 2017. Fake news detection using naive bayes classifier, in: 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), IEEE. pp. 900–903.
[13] Grinberg, N., Joseph, K., Friedland, L., Swire-Thompson, B., Lazer, D., 2019. Fake news on twitter during the 2016 us presidential election. Science 363, 374–378.
[14] Guess, A., Nagler, J., Tucker, J., 2019. Less than you think: Prevalence and predictors of fake news dissemination on facebook. Science advances 5, eaau4586.
[15] Hakim, A.A., Erwin, A., Eng, K.I., Galinium, M., Muliady, W., 2014. Automated document classification for news article in bahasa indonesia based on term frequency inverse document frequency (tf-idf) approach, in: 2014 6th international conference on information technology and electrical engineering (ICITEE), IEEE. pp. 1–4.
[16] Horne, B., Adali, S., 2017. This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news, in: Proceedings of the International AAAI Conference on Web and Social Media.
[17] Huckle, S., White, M., 2017. Fake news: A technological approach to proving the origins of content, using blockchains. Big data 5, 356–371.
[18] Kannan, S., Gurusamy, V., Vijayarani, S., Ilamathi, J., Nithya, M., 2014. Preprocessing techniques for text mining. International Journal of Computer Science & Communication Networks 5, 7–16.
[19] Kao, A., Poteet, S.R., 2007. Natural language processing and text mining. Springer Science & Business Media.
[20] Lazer, D.M., Baum, M.A., Benkler, Y., Berinsky, A.J., Greenhill, K.M., Menczer, F., Metzger, M.J., Nyhan, B., Pennycook, G., Rothschild, D., et al., 2018. The science of fake news. Science 359, 1094–1096.
[21] Lee, N.M., 2018. Fake news, phishing, and fraud: a call for research on digital media literacy education beyond the classroom. Communication Education 67, 460–466.
[22] Luhn, H.P., 1957. A statistical approach to mechanized encoding and searching of literary information. IBM Journal of research and development 1, 309–317.
[23] Pérez-Rosas, V., Kleinberg, B., Lefevre, A., Mihalcea, R., 2017. Automatic detection of fake news. arXiv preprint arXiv:1708.07104 .
[24] Rashkin, H., Choi, E., Jang, J.Y., Volkova, S., Choi, Y., 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking, in: Proceedings of the 2017 conference on empirical methods in natural language processing, pp. 2931–2937.
[25] Rubin, V.L., Chen, Y., Conroy, N.K., 2015. Deception detection for news: three types of fakes. Proceedings of the Association for Information Science and Technology 52, 1–4.
[26] Rubin, V.L., Conroy, N., Chen, Y., Cornwell, S., 2016. Fake news or truth? using satirical cues to detect potentially misleading news, in: Proceedings of the second workshop on computational approaches to deception detection, pp. 7–17.
[27] Schapire, R.E., 1990. The strength of weak learnability. Machine Learning 5, 197–227.
[28] Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H., 2017. Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter 19, 22–36.
[29] Straková, J., Straka, M., Hajic, J., 2014. Open-source tools for morphology, lemmatization, pos tagging and named entity recognition, in: Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 13–18.
[30] Tandoc Jr, E.C., Lim, Z.W., Ling, R., 2018. Defining "fake news" a typology of scholarly definitions. Digital journalism 6, 137–153.
[31] Wang, K., Thrasher, C., Viegas, E., Li, X., Hsu, B.j.P., 2010. An overview of microsoft web n-gram corpus and applications, in: Proceedings of the NAACL HLT 2010 Demonstration Session, pp. 45–48.
[32] Webster, J.J., Kit, C., 1992. Tokenization as the initial phase in nlp, in: COLING 1992 Volume 4: The 15th International Conference on Computational Linguistics.
[33] Zannettou, S., Sirivianos, M., Blackburn, J., Kourtellis, N., 2019. The web of false information: Rumors, fake news, hoaxes, clickbait, and various other shenanigans. Journal of Data and Information Quality (JDIQ) 11, 1–37.