Sentiment Analysis for Fake News Detection by Means of Neural Networks

Sebastian Kula[1,2], Michał Choraś[1], Rafał Kozik[1], Paweł Ksieniewicz[1,3], and Michał Woźniak[1,3]

[1] UTP University of Science and Technology, Bydgoszcz, Poland
[2] Kazimierz Wielki University, Bydgoszcz, Poland skula@ukw.edu.pl
[3] Wrocław University of Science and Technology, Poland

Abstract. The problem of fake news has become one of the most challenging issues having an impact on societies. Nowadays, false information may spread quickly through social media. In that regard, fake news needs to be detected as fast as possible to avoid negative influence on people who may rely on such information while making important decisions (e.g., presidential elections). In this paper, we present an innovative solution for fake news detection that utilizes deep learning methods. Our experiments prove that the proposed approach allows us to achieve promising results.

Keywords: Online disinformation · Fake news · Neural Networks · Deep learning · Sentiment Analysis

1 Introduction

Fake news is often defined as a hoax or false information that is spread employing the news media, either printed or online social networks. This phenomenon is not new in human history, and one can find examples of fake news originating in the nineteenth century (e.g., Great Moon Hoax [1]). However, due to the increasing popularity of social media widely used for political purposes, the problem of fake news has gained more importance in recent years. It also imposes a great detection challenge. Manual fact-checking in many cases, is difficult, time-consuming, and expensive. Therefore, the community has been looking for various automated detection solutions that would speed up this process. In recent years, different NLP (Natural Language Processing) methods have been proposed to solve the fake news detection problem.

The main contribution of this paper is the proposition and evaluation of neural network-based approach to text analysis and fake news detection. The contribution includes the application of the remote, cloud computing platform, GPU cards, state-of-the-art Machine Learning and Deep Learning libraries; all the above-mentioned works allowed to create a working model for fake news detection in a relatively short time and with the use of open-source solutions.

The paper is structured as follows: after the introduction, in Section 2 the proposed approach is presented in detail. In Section 3 the used datasets are

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

overviewed. Experimental setup and results are described in Section 4, whereas conclusions are given after that.

2 Related Work

There are various machine learning approaches for fake news detection and the main effort is put into efficient feature extraction, as well as an appropriate choice of the classification model.

In [2], authors have adopted Naïve Bayes to recognize fake and legitimate news. On the other hand, Shu et al. in [3], explicitly elaborated and listed a set of attributes that may help to indicate fake news. These attributes include the source (author or the publisher of the news), headline (a sort of title that is intended to draw readers' attention), body (the main text describing the news), image or video that is intentionally used for spotting the fake news.

In the literature, some approaches utilise computer vision for fake news detection. An interesting method, falling into that category, for image-based fake photos detection has been presented in [4].

Recently, with the emergence of deep learning, a significant number of researchers have started applying this type of model to solve various classification and regression problems. The deep learning methods are capable of autonomously computing the hierarchical representation of the data and allow achieving results that surpass other state-of-the-art approaches.

Zang et al. [5] have proposed a deep recurrent diffusive neural network to address the problem of fake news detection. On the other hand, in contrast to the traditional RNN model, in [6] authors adapted a pre-trained BERT model (Bidirectional Encoder Representations from Transformers), that consists of several stacked transformer-encoder blocks.

3 The Proposed Approach

To tackle the challenge of fake news, the application of the Flair library is proposed [7], which offers outstanding features in terms of neural network design, includes many state-of-the-art methods, among them numerous methods based on the deep learning, also enabling GPU-based training. Flair is a Natural Language Processing library designed for all word embeddings as well as arbitrary combinations of embeddings [7]. The crucial elements of creating the fake news detection model were carried out with the support of the Flair library. The training process was carried out based on deep learning methods afterword embeddings had been carried out using the modern and effective procedures in this area. As shown in Fig. 1, in our work, we chose to use various types of neural networks to solve the problem of text-based fake news detection.

3.1 Text Pre-processing Using NLP

The goal of text pre-processing is to obtain the text, which is a reduced representation of the raw text. The reduced text enables the detection of specific patterns

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

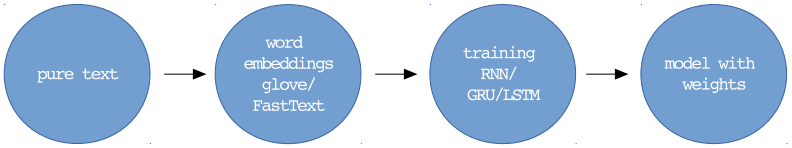

Fig. 1. The processing pipeline of the proposed solution

of the raw text simultaneously. A reduction strategy was adopted, consisting of the elimination of unnecessary elements and, through this step, achieving a higher generalization of the text. To detect unnecessary items and overrepresentation of words, statistical analysis of their occurrences in datasets was used. The clean text was obtained by creating a separate code, unrelated to the Flair library.

Due to Flair use of embedding layers, it is not necessary to run the usual preprocessing steps such as constructing a vocabulary of words in the dataset or encoding words as one-hot vectors [7]. In the Flair, each embedding layer implements either the TokenEmbedding or the DocumentEmbedding interface for word and document embeddings respectively [7]. In our approach, we treated the content of articles as documents and we applied the DocumentEmbedding interface.

3.2 Word Embeddings in Flair

Neural networks used in NLP tasks do not operate directly on texts, sentences, or words, but on their representation in the numerical form. This process of converting them into numbers is called word embeddings and it is one of the key elements enabling sentiment analysis and fake news detection.

The main methods of word embeddings are 'word2vec', 'glove', and 'FastText', which are classified as canonical methods. In addition to the listed above, the Flair library supports a growing list of embeddings such as hierarchical character features, ELMo embeddings, ELMo transformer embeddings, BERT embeddings, byte pair embeddings, Flair embeddings and Pooled Flair embeddings [7].

In this work, the 'glove' method was used. For comparative purposes, the 'twitter' word embeddings, 'news' word embeddings and 'crawl' word embeddings were used as well. The synthesis of the methods used is summarized below.

The 'glove' is an open-source project at Stanford University; its code is freely available [8]. The 'glove' overcomes the disadvantages of the models focusing only on local statistics and the models focusing only on global statistics. For example, methods like latent semantic analysis (LSA) efficiently leverage statistical information, but they do relatively poorly on the word analogy task [8]. The other example, skip-gram methods, may do better on the analogy task, but they poorly

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

utilize the statistics of the corpus [8]. The 'glove' is a specific weighted least squares model that trains on global word-word co-occurrence counts and thus makes efficient use of statistics [8]. The 'glove' is a global log-bilinear regression model for the unsupervised learning of word representations that outperforms other models on word analogy and word similarity [8].

To launch word embeddings in the Flair with the use of 'glove', the user enters the following WordEmbeddings('glove') command in the code.

The FastText method was created by the Facebook AI Research lab based on the models contained in the article [9]. The FastText method is based on the bag of n-grams and subword units. Each word is represented as a bag of character n-grams [10]. The method indicates better results than the state-of-the-art methods in word similarity and word analogy experiments [10].

Both methods are available in the Flair library as pre-trained databases of word embeddings. The 'glove' and the FastText methods were created based on data obtained from Wikipedia. Pre-trained models used in this paper, like 'news', were created using FastText embeddings over news and Wikipedia data; the 'crawl' was created using the FastText embeddings over web crawls; 'twitter' was created using two billion tweets.

3.3 Recurrent Neural Network

Currently, the text classification methods most often use methods based on Deep Neural Networks (DNN), which have better performance in Natural Language Processing (NLP) tasks solving than other neural networks. DNNs are characterized by high complexity and a large number of hidden layers, which is their distinguishing feature in comparison with standard Artificial Neural Networks (ANN).

Deep Neural Networks have already been extensively used in many areas of artificial intelligence, such as speech recognition, image recognition, text translation, sentiment analysis, and spam detection. There is a whole range of DNN methods used in NLP. In this article, we focus on Recurrent Neural Network (RNN) as well as Gated Recurrent Unit (GRU) and Long-Short Term Memory (LSTM) methods that are classified as RNN methods (networks).

The feature distinguishing RNN networks from other ANN networks is their recurrency, referring to the flow of signals between input and output of the network. This type of networks has a kind of feedback loop, which means that the output is also the input for the next state and affects its output value. Such a network architecture results in the fact that the network has a kind of memory that theoretically allows for information storage. Apart from the difference mentioned above, the RNN network works like a regular, one-way ANN network, that is during the training weights and propagation errors are calculated.

The disadvantage of the RNN network is the phenomenon of the vanishing gradient, which makes it impossible to remember and search for the bindings between data that occur after a more extended period. There are several meth-

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

ods that overcome this undesirable phenomenon; they include GRU and LSTM networks. Both networks are described in the next two subsections.

The GRU is a network with recursive architecture, improved by introducing special types of gates: the reset gate r_t and the update gate z_t. They together control how information is updated to the state [11]. In mathematical terms, the z_t gate is as follows [11]:

$$z_t = \sigma(W_z x_t + U_z h_{t-1})$$

where σ is sigmoid function, W_z and U_z are weights, h_{t-1} is previous state, x_t is the sequence vector at time t. The formula for r_t gate is as follows [11]:

$$r_t = \sigma(W_r x_t + U_r h_{t-1})$$

The formula is similar to the formula for the update gate, with the noticeable different weights W_r and U_r. The candidate activation h'_t is computed similarly to that of the traditional recurrent unit [11] [12]:

$$h'_t = tanh(W_h x_t + r_t \odot U_h h_{t-1})$$

where r_t is a set of reset gates and $\odot$ is an element-wise multiplication [12]. When the reset gate is zero, the previous computed state is erased from the network. The activation h_t of the GRU at time t is a linear interpolation between the previous activation h_{t-1} and the candidate activation h'_t [11] [12]:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot h'_t$$

The neural network constructed in this way allows to control and collect the data and thus to tackle the issue of the vanishing gradient.

The LSTM is a neural network that is similar to the GRU, except that it is more complex and requires more computing power during training. The LSTM contains the following gates: input, output, forget, memory cell and new memory cell content [12]. The forget gate determines whether the memory cell content will be preserved or erased; the input gate determines whether the new memory cell content will be added to the memory cell; the output gate decides what content from the memory cell will be on the output. There are many versions of the LSTM implementation; the original one was presented in the article [13].

4 Experimental Evaluation

The purpose of the experiment is to conduct the training with the data and then to validate the model using the Flair library. The application of the Flair library to create the fake news detection model has not been reported in the literature yet.

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

4.1 Experimental Setup

Finding the right dataset is fundamental to create an efficient, reliable fake news detection model. Simultaneously, the access to such datasets is limited and creates a challenge to acquire current, ready-to-learn databases. In the article, we applied freely available datasets, which are accessible on the websites of Kaggle and the Information Security and Object Technology (ISOT) research lab. Two different sets of data were applied, one called "ISOT Fake News Dataset" [14], the other called "Getting real about fake news"(GRaFN) [15].

Two models were taught, the first one is based on the application of the ISOT dataset for training, and the second model, because the collection acquired from the Kaggle contains mostly fake news, was taught with the use of both collections, through attaching the real news collection from ISOT to the collection downloaded from the Kaggle webpage.

Information in both datasets contains news published on websites. The ISOT collection is dominated by the vast majority of political information and news from around the world. The dataset contains two files (true and fake) in csv file format. The real information database was created based on the websites of a reliable Reuters news agency, and the fake information was collected from the pages marked as unreliable by Politifact [16]. The dataset contains a total of 44898 items, 21417 of which are real items and 23481 are fake items. Each file contains four columns: article title, text, article publication date and the subject which can relate to one of six types of information (world-news, politics-news, government-news, middle-east, US news, left-news) [14]. In order to prepare the data for pre-processing in a proper way, an analysis of the occurrence of e-mail addresses, social media addresses, website addresses (https and www) was conducted, the results are presented in Table 1.

Table 1. The analysis of the occurrence of email addresses, social media addresses, website addresses (https and www) in the ISOT dataset

Type of address	ISOT Dataset (True)	ISOT Dataset (Fake)
email and social media addresses	803	27888
https addresses	0	94
www addresses	48	726

In the file with true data, 28909 instances of Reuters information agency name were observed; such a large number of instances can become a special feature of the pattern, hence we decided to eliminate this property from the data. The number of occurrences of city names was also analysed to determine the nature of the information in geographical terms. Table 2 presents the 10 most common city names in the items.

The second dataset contains texts classified into eight categories (bias with 443 occurrences, conspiracy - 430 occurrences, fake - 19 occurrences, bs - 11492 occurrences, hate - 246 occurrences, junksci - 102 occurrences, satire - 146 oc-

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

Table 2. The 10 most common city names occurrence in the ISOT dataset

Name of the city	ISOT Dataset (True)	ISOT Dataset (Fake)
WASHINGTON	6921	107
NEW YORK	949	16
LONDON	761	3
MOSCOW	673	1
BERLIN	530	2
BEIJING	503	0
PARIS	331	13
ANKARA	275	0
MEXICO CITY	237	4
TOKYO	228	0

currences and state - 121 occurrences); in this paper, all these categories but satire can be considered to be various forms of fake news.

The dataset contains 20 columns, 12999 rows containing the same number of articles in the text column. The dataset includes the text from 244 websites, collected within 30 days [15]. Most of the articles were written in English (12403 articles), but there are also articles in Russian (203 articles), Spanish (172 articles), German (111 articles), French (38 articles), Arabic (22 articles), Portuguese (11 articles), Turkish (10 articles), Italian (9 articles) or Chinese (1 article). Only the English-language articles were used to create the model. Similarly to the first dataset for the correct pre-processing, the number of e-mail addresses occurrences, social media addresses occurrences, website addresses (https and www) occurrences, emoticons occurrences were detected; data in Table 3.

Table 3. The analysis of the occurrence of email addresses, social media addresses, website addresses (https and www) and emoticons in the "Getting real about fake news" dataset

Type of string	Number of occurrences in GRaFN dataset
email and social media addresses	3512
https addresses	25
www addresses	1499
emoticons	93

The described datasets were used not only for the training process but also for validation and testing of neural network models. The initial step to create the model was to prepare the data by pre-processing them. The data underwent many handlings consisting of the elimination of unnecessary elements or the ones disturbing the training process and simultaneously occurring in significant amounts in the data. First, datasets were limited to two key columns, the *label* column, determining if the information is true or false, and the *text* column, containing the contents of the articles. Then, all the items beyond stan-

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

dard text were eliminated from the articles in the *text* column. Social media addresses, e-mail addresses, website addresses (https and www), emoticons and even punctuation and periods were removed.

Two data collections were used to conduct the experiments. Collection 1 is based entirely on the ISOT dataset and, according to the cross-validation procedure implemented in the Flair, contains three elements: data for the training, for the testing, and the validation. To ensure adequate representativeness of the data in the collection 1, the ISOT dataset was divided into three parts, in the following proportion: 80% allocated for the training, 10% for the testing and 10% for the validation. Collection 2 was created from the combination of the extracted part of the ISOT dataset, containing real news, and the GRaFN dataset. As in the collection 1, data representativeness in collection 2 is ensured by dividing into three parts in the same proportion as for the collection 1.

Pre-processing and training operations followed by post-processing were carried out in the cloud service environment called the Colaboratory. The pandas library version 0.25.3, the Flair library version 0.4.3, Jupyter notebook and the Python programming language were used. In the hardware scope, the hardware resources available on the Colaboratory platform were used, in the form of the GPU card P100 PCIE-16 GB, cuda version 10.1, 12.72 GB RAM memory, 68.4 GB HDD memory.

Before the training, in addition to preparing the data and hardware, the neural network architecture needs to be designed and its hyperparameters need to be specified. The Flair library allows the user to specify the values of many hyperparameters; some selected ones are presented in Table 4. All created models had exactly the same hyperparameters values set before the training.

Table 4. Hyperparameters values of RNN GRU

Name of the hyperparameter	Hyperparameter value
learning rate	0.1
batch size	32
anneal factor	0.5
patience	5
max number of epochs	5
hidden states size	512

A crucial element in machine learning is the correct selection of metrics. The classification tasks focus most often on the following metrics: accuracy, f1 score, precision and recall. The paper presents the analysis of accuracy, precision and recall during the tests and and f1 score during the validation processes. The accuracy was defined as a number correct predictions divided by the total number of predictions, multiplied by 100%. The f1-score was defined according to the following formula:

$$f1score = \frac{2 * precision * recall}{precision + recall}$$

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

where precision and recall are validation values.

4.2 Plan of the Experiment

The experiment involved three different routines (training and validation) for generating models and tests of the created models. The routine 1 consisted of using the collection 2, the LSTM neural network and word embeddings 'glove', the routine 2 consisted of using the collection 1, the neural network GRU and the word embeddings 'glove', the routine 3 was the extension of the routine 2 by adding parameterization, consisting of changing the word embedding methods from 'glove' to 'news', 'twitter' and 'crawl'. The goal of the tests was to demonstrate the usability of the created models in practice.

4.3 Results and Discussion

When analyzing the results, the focus was on examining the training loss, the validation loss, the f1 score parameter, the accuracy and the computation time needed to train the neural network. The results are shown in Figs. 2-8. Based on the analysis of losses, it can be concluded that, apart from the case presented in Fig. 5 for the 'twitter' method, there is no risk of overfitting the network, because the training loss curve usually has lower values or relatively slightly higher values than the validation loss curve. For all methods, a high f1 score was obtained - above 0.9 for the epoch 5. In Flair, the f1 score metric is crucial in the validation process. On its basis, the best model is automatically selected for tests. It is observed that not always the best result and the best model is obtained for the last epoch. For example, for the 'twitter' method the highest f1 score was obtained for the fourth epoch.

All trained models have obtained sufficient testing accuracy to be applied in the practical tasks of fake news detection. Based on a dataset with similar topics as the ISOT dataset, which is politics-related topics, in [14] for the fake news detection challenge, the authors obtained a maximum accuracy of 92%. In all our experiments, the maximum obtained accuracy outperformed this value.

Table 5. Resulted metrics for testing of models for the label fake (the comparison between word embeddings techniques, 'glove','news', 'twitter', 'crawl')

Metric	'glove'	'news'	'twitter'	'crawl'
true positives (TP)	2275	2080	2269	2217
true negatives (TN)	2139	2121	2064	2100
false positives (FP)	4	22	79	43
false negatives (FN)	2	197	8	60
precision	0.9982	0.9895	0.9664	0.9810
recall	0.9991	0.9135	0.9965	0.9736
accuracy	99.86%	95.04%	98.03%	97.67%

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

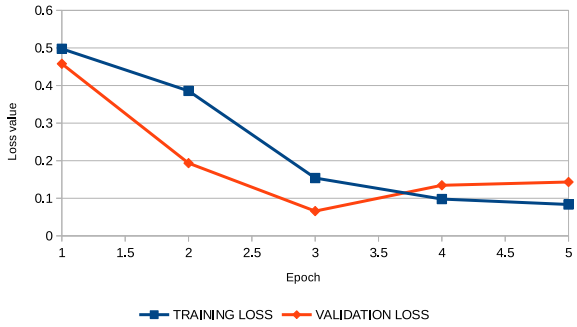

Fig. 2. Relations between the number of epochs versus the training and validation losses; the results obtained during the training of the model with the use of both dataset collections, and the word embeddings technique 'glove'

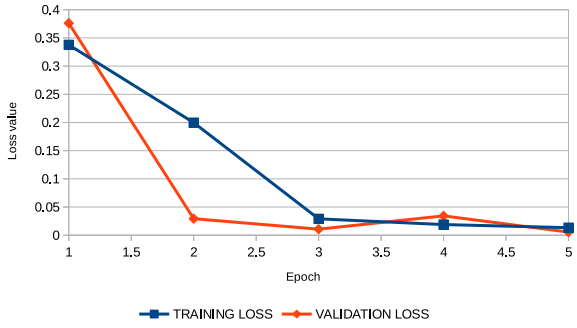

Fig. 3. Relations between the number of epochs versus the training and validation losses; the results obtained during the training of the model with the use of the ISOT dataset collection, and the word embeddings technique 'glove'

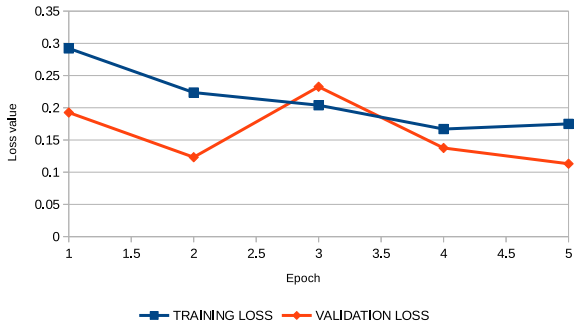

Fig. 4. Relations between the number of epochs versus the training and validation losses; the results obtained during the training of the model with the use of the ISOT dataset collection, and the word embeddings technique 'news'

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

Fig. 5. Relations between the number of epochs versus the training and validation losses; the results obtained during the training of the model with the use of the ISOT dataset collection, and the word embeddings technique 'twitter'

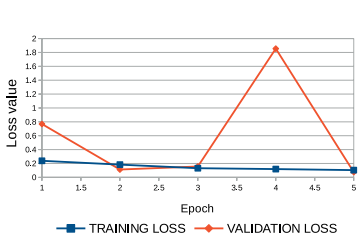

Fig. 6. Relations between the number of epochs versus the training and validation losses; the results obtained during the training of the model with the use of the ISOT dataset collection, and the word embeddings technique 'crawl'

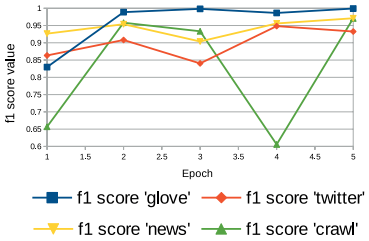

Fig. 7. Analysis of f1 score; the comparison between word embeddings techniques

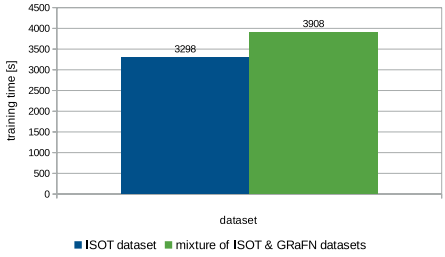

Fig. 8. Computation time needed for models training; the comparison between datasets applied for the training

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

As a result of the training, models with calculated parameters, i.e., weights were obtained. These models were tested on the remote Colaboratory platform. The first stage of testing consisted of entering selected texts contained in the ISOT dataset. In the second stage of testing, the Independent [17], which is a British online newspaper, and the 11 Sci-Fi Short Stories website [18] were used. Articles in the Independent were treated as a source of credible texts (true), and science fiction stories as a source of unreliable materials (false). For both the above-described tests, the model worked correctly and detected true and false information. Tests were carried out repeatedly, confirming the validity, robustness and credibility of the model. Examples of correct model operation are shown in Fig. 9. All entered texts were subjected to the procedure of eliminating irrelevant elements from texts, before submitting them to the model. The procedure is identical to the one carried out in the pre-processing stage on the raw data.

```
# create example sentence
sentence = Sentence('The space station staff liked her when they interviewed her she seemed
polite and quiet and incurious. That was important. One of the astronauts, a bearded
Russian with kind eyes, asked her a question: Will you be lonely in space? She looked at
the faint lines scrawled around his eyes and forehead, and she supposed he had a family
somewhere, maybe small children. Yes, she said, but I have always been lonely. The
astronaut nodded, and she could see he understood. She could see his aquiline profile as he
turned to someone off screen, and she knew she would get the job.')



# predict tags and print
classifier.predict(sentence)

print(sentence.labels)
```

```
__label__fake (0.86463862657547)
```

Fig. 9. The screenshot of the launched model, which recognizes that the information from the 11 Sci-Fi Short Stories website is fake

The model can be used in real-time solutions because its execution time is relatively short. The results indicated the impact of word embedding techniques on the accuracy and the f1 score. The highest results were obtained for the 'glove' method. In the process of creating the model and training the neural network, it was observed that one of the crucial elements to obtain robust results is the correctly performed pre-processing on the raw data. The relatively short time needed to train the neural network was achievable by applying for the GPU card.

4.4 Threats to Validity

The proposed method should be tested on other datasets. The critical and most desired scenario would be to have fake and true news obtained from the same source/ news agency. In our experiments, true news is not from the same source as fake news. However, such datasets are still to be offered by reliable news agencies (which most often claim not to have fake news at all).

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

Another aspect is the lack of a clear definition of what is exactly meant by fake news. For example, conspiracy theories type of information is not always considered as fake news. In such definitions, the motive of the source is taken into account, but it cannot be determined only by analyzing the text as we do in this work.

5 Conclusions

The paper presents the stages of creating the model applied for fake news detection. The model is based on DNN networks trained with the Flair library. The pre-processing, training and post-processing phases are described in detail for the obtained models. The novelty of the paper is the application of the Flair library for detecting true and false information, as well as the application of the cloud solution called the Collaboratory. The model fulfills its tasks and allows for the analysis of texts with high accuracy. During the training process, the accuracy was up to 99.8%.

The current work concerned the distinction between label fake and label true. However, there are many additional subcategories under the fake news category; future work will concern the creation of a model to distinguish those sub-categories.

Acknowledgement

This work is funded under SocialTruth project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 825477.

References

1. Goodman, Matthew, The Sun and the Moon: The Remarkable True Account of Hoaxers, Showmen, Dueling Journalists, and Lunar Man-Bats in Nineteenth-Century New York, New York: Basic Books, 2008
2. A. Jain and A. Kasbe, "Fake News Detection," 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, 2018, pp. 1-5.
3. Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, Huan Liu, Fake News Detection on Social Media: A Data Mining Perspective, CoRR, abs/1708.01967, 2017
4. Chorać M., Giełczyk A., Demestichas K., Puchalski D., Kozik R. (2018) Pattern Recognition Solutions for Fake News Detection. In: Saeed K., Homenda W. (eds): CISIM 2018. LNCS vol. 11127. Springer, Cham
5. Jiawei Zhang, Limeng Cui, Yanjie Fu, Fisher B. Gouza. Fake News Detection with Deep Diffusive Network Model, CoRR, abs/1805.08751, 2018
6. Devlin, Jacob and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv preprint arXiv:1810.04805, 2018

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49

7. Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., Vollgraf, R.: FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP. In: Ammar, W., Louis, A., and Mostafazadeh, N. (eds.) NAACL-HLT (Demonstrations). pp. 54–59. Association for Computational Linguistics (2019).
8. Pennington, J., Socher, R., Manning, C.D.: GloVe: Global Vectors for Word Representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014).
9. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of Tricks for Efficient Text Classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. pp. 427–431 (2017).
10. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching Word Vectors with Subword Information. Transactions of the Association for Computational Linguistics. 5, 135–146 (2017).
11. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical Attention Networks for Document Classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1480–1489 (2016).
12. Chung, J., Gülcehre, C., Cho, K., Bengio, Y.: Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. In: NIPS 2014 Workshop on Deep Learning and Representation Learning. , Montréal, Canada (2014).
13. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation. 9, 1735–1780 (1997).
14. Ahmed H, Traore I, Saad S.: Detecting opinion spams and fake news using text classification. Journal of Security and Privacy **1**(1), e9 (2018)
15. Getting real about fake news, https://www.kaggle.com/mrisdal/fake-news. Last accessed 25 Nov 2019
16. Ahmed, H., Traoré, I., Saad, S.: Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. In: Traoré, I., Woungang, I., and Awad, A. (eds.) ISDDC. pp. 127–138. Springer (2017).
17. The Independent webpage, https://www.independent.co.uk/news/world/middle-east/raqqa-isis-terror-fears-europe-a7401511.html. Last accessed 28 Dec 2019
18. 11 Sci-Fi Short Stories, the Janitor in Space - American Short Fiction website, http://americanshortfiction.org/2014/07/01/janitor-space/. Last accessed 28 Dec 2019
19. Ksieniewicz P., Choraś M., Kozik R., Woźniak M., Machine Learning Methods for Fake News Classification , in: Yin H., Camacho D., Tino P., Tallón-Ballesteros A., Menezes R., Allmendinger R. (eds) Intelligent Data Engineering and Automated Learning – IDEAL 2019. LNCS vol. 11872, 332-339, Springer, 2019.
20. Choraś M., Pawlicki M., Kozik R., Demestichas K.P, Kosmides P., Gupta M., SocialTruth Project Approach to Online Disinformation (Fake News) Detection and Mitigation , In Proc. of ARES 2019, 68:1-68:10, Canterbury, UK, 2019.

ICCS Camera Ready Version 2020
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-50423-6_49