Fake news detection using deep learning

Conference Paper · December 2020

3 authors:

Imane Ennejjai
Mohammed V University of Rabat
7 PUBLICATIONS **5** CITATIONS
SEE PROFILE

Sara Ibn El Ahrache
Ecole de Sciences Appliqués de Tanger
17 PUBLICATIONS **41** CITATIONS
SEE PROFILE

Badir Hassan
Abdelmalek Essaâdi University
128 PUBLICATIONS **418** CITATIONS
SEE PROFILE

Fake news detection using deep learning

Imane Ennejjai[1], Sara Ibn Ahrache[2], and Hassan Badir[3]

Data engineering and system Team (IDS) - Abdelmalek Essaadi University, National School of Applied Sciences of Tangier , Morocco
imane.ennejjai@etu.uae.ac.ma
sara.elahrache@gmail.com
badir.ensa@gmail.com

Abstract. The detection of fake news was treated as a text classification problem. Over a hundred experiments were performed to find an appropriate combination of preprocessing and efficient neural network architecture, underlining some specificities and limitations of the Fake News detection problem over other research tasks. Different automatic learning approaches have been tried to detect it. However, most of these focused on a particular type of news and did not apply many advanced tech-niques. The objective of this thesis is to evaluate the performance of different approachesbased on features of Neuro linguistic programming over supervised learning. The models are examined against four datasets, one containing online news articles and theother information from various sources

Keywords: Fake news · nlp · Disinformation · Text Classification · covid19.

1 Introduction

Detecting fake news involves categorizing news based on its veracity. In a simple context, this is a binary classification task, while in a more difficult context, it is a fine classification task. Detecting fake news has been one of the hottest research topics in artificial intelligence recently. Due to the availability of the internet and the willingness to share information through social media, it is easy to make fake news and spread it around the world. When disseminated widely, fake news can have a dramatic negative impact on many aspects of life. To date, there are a variety of approaches to fake news detection [11]. Despite considerable crowd attention, detection of fake news has not progressed much for some time due to insufficient data on fake news. In this work, our objective is to present a comparative analysis of the performances of existing methods by implementing each on four available datasets. We are also integrating different functionalities of existing works and studying the performance of some successful text classification techniques. Specifically, in this work, we present the performance of traditional models of machine learning and deep learning on four datasets that contain news on multiple topics. We thus present the performances of some advanced models such as convolutionnal-HAN, Biderctionnel LSTM.

2 Related work

Work on fake news detection is almost non-existent and focuses mainly on the 2016 US presidential elections or does not use the same features. That is, when this work focuses on automatic feature extraction using machine learning and deep learning, other work uses artisanal features such as psycholinguistic features that are not the goal in our project. Current research focuses primarily on the classification of online news and social media posts. Different methods have been proposed by different research for the detection of deception. Shu, Silva, Wang, Jiliang and Liu [1] proposed to use linguistic characteristics such as total number of words, characters per word, frequencies of large words, frequencies of sentences, parts of speech markup (POS). They took an in-depth look at fake news detection on social media, from the perspective of data mining, evaluation metrics, and representative datasets. Wang compared the performance of SVM, LR, Bi-LSTM, CNN models on their proposed Liar dataset [2]. Several research works show promising results in detecting fake news via the neural network and tracking user propagation. Wang in his [2] constructed a hybrid convolutional neural network model that outperforms other traditional machine learning models. Ruchansky et al. [4] proposed a deep hybrid model for fake news detection using several types of features such as time engagement between n users and m news articles over time and produced a label for the categorization of fake news, but also a score for suspicious users.they used an RNN to extract temporal characteristics from news content and a fully connected network in the case of social features. Tacchini et al.[5] focus on using social media features to improve the reliability of their detector. They used logistic regression and the harmonic algorithm[6] to classify the information into hoax and non-hoax categories. Harmonic Algorithm is a method for transferring information between users who liked some common messages. The problem was approached by Bajaj[9] from a pure NLP perspective using convolutional neural networks (CNN). Ahmed et al.[7] introduced a new n-gram model for automatically detecting bogus content, with a particular focus on reviews and news. Results of two techniques for extracting different characteristics, viz. tf, tf-idf, and six machines learning classification techniques have been reported by the authors. Pérez-Rosas et al.[8] introduced two new datasets for the fake news detection task, covering seven different news areas. From a set of learning experiences to detect fake news, the authors concluded that accuracies of up to 76% could be obtained.Reis et al.[3] use machine learning techniques on the US election buzzfeed article. The algorithms evaluated are k-Nearest Neighbors, Naive-Bayes, Random Forests, SVM with RBF kernel and XGBoost. Their results show that XGBoost is good at selecting texts that need to be verified manually, this means that texts classified as reliable are indeed reliable, and therefore reduce the amount of texts to be verified manually. This model is limited by the fact that they use metadata that is not always available. Arvinder et al.[10] demonstrates with extensive experimentation from the point of view of natural language processing and machine learning; Assessment is performed for three standard data sets with a new set of additional headings and content characteristics. Khan et al.[11] specifically compares traditional algorithms such as logistic regression, support vector machine, decision trees, naive bayes, and neighbor K-Nearest over a wide range of neural architectures based on CNN or LSTM. The naive bayes classifier works surprisingly well, while the performance of neural networks depends on expanding the underlying dataset. The most of previous work on fake news detection has applied several traditional methods of machine learning and neural networks to detect fake news. However, they focused on detecting information of particular types (such as policies). As a result, they developed their models and functionality for specific data sets that match their topic of interest. It is likely that these approaches suffer from a bias in the datasets and are likely to perform poorly on news from another topic. Some of the existing studies have also made comparisons between different methods of detecting fake news. A major limitation of previous comparative studies is that as these are conducted on a specific type of data set, it is difficult to reach a conclusion about the performance of various models. Additionally, these articles focused on a limited number of features that resulted in an incomplete exploration of the potential characteristics of fake news. We have seen previously that most of the related work focuses on improving the quality of the prediction by adding additional features. The point is that these features are not always available, for example some articles may not contain images. There is also the fact that the use of information from social media is a problem because it is easy to create a new account on this media and to trick the detection system. That's why we've chosen to focus solely on the body of the article and see if it's possible to accurately detect fake news. As has been shown in the previous sections, there are several approaches that can be used to extract features and use them in models. This focuses on the functionality of textual news content.

3 Overview of the approach

Our general approach to fake news detection is shown in Figure 1. It begins by preprocessing the data set, removing unnecessary characters and words from the data. N-gram entities are extracted and an entity matrix is formed representing the documents involved. The last step in the classification process is to train the classifier.

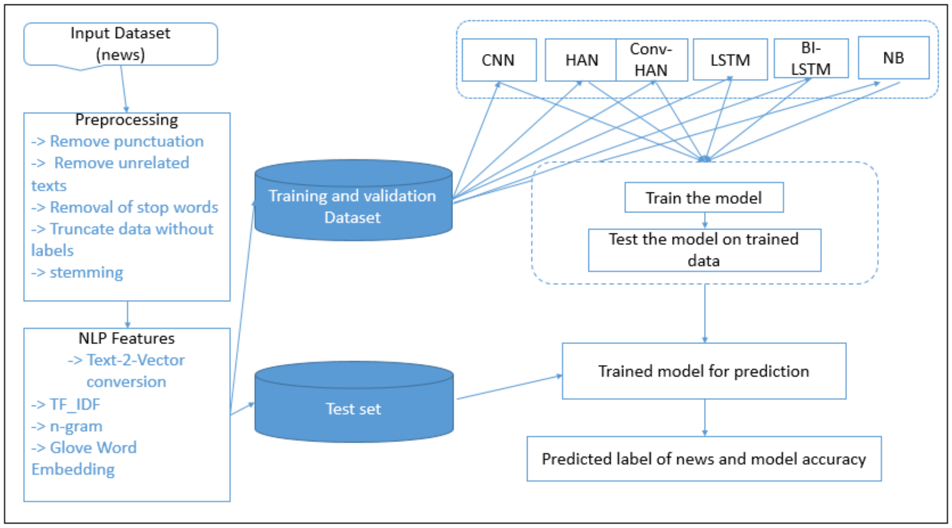
Input Dataset (news)
Preprocessing
-> Remove punctuation
-> Remove unrelated texts
-> Removal of stop words
-> Truncate data without labels
-> stemming
NLP Features
-> Text-2-Vector conversion
-> TF_IDF
-> n-gram
-> Glove Word Embedding
Training and validation Dataset
Test set
CNN
HAN
Conv-HAN
LSTM
BI-LSTM
NB
Train the model
Test the model on trained data
Trained model for prediction
Predicted label of news and model accuracy

Fig. 1. Flowchart of the proposed fake news detection process

4 Experimental Evaluation

4.1 Datasets Statistical Information

The models were tested on four different data sets: (i) Data an Open Sources dataset containing 9,408,908 articles from which 11,161 articles of false and reliable categories were selected. (ii) The Fake or Real News dataset is developed by George McIntire. The fake news portion of this dataset was collected from Kaggle's fake news dataset including the news of the 2016 election cycle in the United States. (iii) Liar is a dataset accessible to the public [23]. It includes 12.8K of short human statements from POLITIFACT.COM API5. (iiii) True / fake dataset is a set of data accessible to the public. The fake part contains 17903 news. the true part contains 20826 news from two labels news politics and wordnews. these data taken from the news between January 13, 2016 to December 31, 2018.

4.2 Features

For building a Deep Learning model, feature selection is of utmost importance for optimum performance of the system. Features used in the proposed model are as follows:

Pre-training word embedding Word embedding is a class of approaches for representing words and documents using dense vector representation. This is an improvement over traditional word bag pattern coding schemes where large, scattered vectors were used to represent each word or to mark each word in a vector to represent an entire vocabulary. Two examples of how to learn to integrate words from text:Word2Vec and Glove In addition to these carefully designed methods, word embedding can be learned as part of a deep learning model. This can be a slower approach, but fits the model to a set of specific training data.

- Glove : GloVe is an unsupervised learning algorithm that allows you to discover the proximity of two words, with their separation in vector space. These created vector representations are called word embedding vectors.
- word2vec : Word2Vec is available in two modes: continuous bag of words (CBOW) and skip gram. It was originally designed to predict a word in a context. For example, given two previous words and the next two words, which word is most likely to occur between them.

n-grams Count Feature These features are used for counting occurrences of n-grams in the title and body of the news, and various ratios of the unique n-gram and total word count given by Eq. (1).

$$ratioofuniquengram = \frac{totaluniquengram}{totalngram} \tag{1}$$

Bag of Words The word bag technique (BoW) treats each news item as a document and calculates the number of frequencies of each word in that document, which is then used to create a digital representation of the data, also known as fixed-length vector features. Bag of Words converts plain text to word count vector with the CountVectorizer function for feature extraction.

tf-idf: Term Frequency- Inverse Document Frequency TF-IDF is an analysis method that can be used in an SEO strategy to determine the keywords and terms that increase the relevance of published texts and therefore of the Web project in its together. It is a formula in which the two values TF (Term Frequency) and IDF (Inverse Document Frequency) are multiplied between them.

5 Features extraction and Model Implementation

5.1 Preprocessing

Textual data requires special preprocessing to implement itmachine learning or deep learning algorithms. There are various techniques widely used to convert textual data into a form ready for modeling. The data preprocessing steps we describe below are applied to the news content. We also provide information on the different representations of word vectors that we used in the framework of our analysis.

word cloud : Before embarking on preproscessing, we visualize our data from the word cloud of the most used keywords in our data.

Punctuation Removal : Natural language punctuation provides the grammatical context for the sentence. Punctuation marks such as a comma may not add much value to understanding the meaning of the sentence.

Stop word removal : We start by removing stop words from the available text data. Stop words are insignificant words in a language that will create noise when used as features in text classification. We used the Natural Language Toolkit - (NLTK) library to remove stop words.

Stemming : is a technique for removing prefixes and suffixes from a word, ending with the root. Using the root, we can reduce the inflectional forms and sometimes the derivative forms of a word to a common base form.

5.2 Feature extraction

The performance of deep learning models depends in large part on the design of the features.

Extraction of n-gram features : Word-based n-gram was used to represent the context of the document and generate functionality to classify the document as false and real. Many existing works have used unigram (n = 1) and bigram (n = 2) approaches for the detection of false news [11].

Pre-trained Word Embedding : For neural network models, word embeddings were initialized with pre-trained 100-dimensional embeddings from GloVe [24]. It was trained on a data set of one billion tokens (words) with a vocabulary of 400 thousand words.

Bag of Words (Bow) : The word bag technique treats each news item as a document and calculates the number of frequencies of each word in that document, which is then used to create a digital representation of the data, also known as fixed-length vector features.

5.3 Implementation of approaches

In this section, we describe the experimental setup of different models based on neural networks and deep learning used in our experiment. We also provide implementation details of our new explored approaches in fake news detection.

CNN The Convolutional Neural Networks model was initialized as a sequence of layers. we will use a fully connected network structure with three layers. Fully connected layers are defined using the Dense class.

LSTM + gensim We use the gensim library in python which supports a bunch of classes for NLP applications. As discussed, we use a CBOW model with negative sampling and 100-dimensional word vectors.

Lstm + glove The LSTM model has been pre-trained with GloVe embeddings in 100 dimensions. The output dimension and time steps were set to 300.

Bi-LSTM The purpose of the Bi-LSTM model is to detect anomaly in a certain part of the news, we need to examine it with both previous and following action events. Bi-LSTM was initialized with pre-trained 100-dimensional GloVe embeddings. An output dimension of 100 and time steps of 300 have been applied.

HAN The hierarchical attention network consisted of two attention mechanisms for word level and sentence level coding. Before training, we set the maximum number of sentences in a press article to 20 and the maximum number of words in a sentence out of 100. In both encoding levels, a two-way GRU with an output dimension of 100 has been introduced into our custom attention layer. We used a word encoder as the input to our sentence encoder time distributed layer.

Convolutional HAN In order to extract high level input characteristics, we have incorporated a one-dimensional convolutional layer before each two-way GRU layer in HAN. This layer selected the characteristics of each trigram from the news article before passing it on to the attention layer.

Naives Bayes We also explored traditional models of machine learning using NLP techniques. We remove suffices from words by deriving them with Snowball Stemmer from NLTK Library.We have introduced the n-gram features there. We used the python library function named MultinomialNB for this.

6 Result

In this section, we describe an analysis of the performance of our neural network-based on deep learning models. We present the best performance for each dataset.We calculate the accuracy, precision, recall and f1 score for the false and real classes, and find their average, media-weighted (the number of true instances for each class) and report an average score of these metrics.

6.1 Evaluation metrics

We use accuracy, precision, recall and f1 as evaluation metrics (tp, fp, fn in the following equations are true positive, false positive and false negative respectively). Precision is a measure calculated as the ratio of correct predictions to the total number of examples. Precision is measuring the percentage of positive predictions that are correct and is defined as:

$$Precision = \frac{tp}{tp + fp} \tag{2}$$

Recall consists of measuring the percentage of correct predictions that the classifier captures and is defined as follows:

$$Recall = \frac{tp}{tp + fn} \tag{3}$$

F1 is to find the balance between recall and precision and is calculated as follows:

$$F1score = \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

		Datasets															
		True / fake				Train / test				Data news				Fake or real news			
models	Features	Ac c	Pre	Rec	F1-	acc	Pre	Rec	F1-	acc	Pre	Rec	F1-	acc	Pre	Rec	F1-
LSTM	word2vec (Genism)	.98	.98	.98	.98	.56	.78	.50	.36	.96	.96	.96	.96	.86	.86	.86	.86
LSTM	Glove Embedding	.98	.98	.98	.98	.54	.27	.50	.35	.75	.75	.75	.75	.74	.75	.74	.74
Bilsm		.78	.50	.78	.89	.61	.61	.59	.59	.84	.84	.84	.84	.84	.84	.84	.84
HAN		.59	.59	.59	.59	.57	.57	.57	57	.94	.94	.94	.94	.86	.86	.86	.86
Conv-HAN		.56	.56	.56	.56	.59	.59	.59	.59	.83	.83	.83	.83	.80	.82	.80	.80
CNN	TF-idfCountvectorizer	.90	.90	.90	.90	.97	.97	.97	.97	.77	.83	.78	.76	.79	.84	.80	.79
Naives –bayes		.97	.97	.97	.97	.83	.87	.84	.83	.97	.97	.97	.97	.90	.90	.90	.90

TABLE 1- Results of the predictive models on the four datasets

6.2 Result and discussion

As noted in Table 1, the model based on attention mechanism are the most vulnerable to overfitting on the True / Fake and Liar dataset. Although BI-LSTM is also victims of overfitting on all datasets and shows their best performance on the True / Fake dataset. Models successfully used for text classification like LSTM, Bi-LSTM, HAN, Conv-HAN hardly overcome the overfitting problem for the Liar dataset. CNN and Naives Bayes models show the best performance among models with 80% accuracy. NB model achieves over 90% accuracy and F1 score over 0.97. This result indicates that although models based on neural networks may suffer from overfitting for a small data set (LIAR).
The Naive Bayes model (with n-gram) has shown the best performance among traditional machine learning models, while CNN, BiLSTM, and Conv-HAN are the most promising among the NN-based models (table). We can see that the n-gram features are very promising in spam detection [13]. Therefore, models based on neural networks may show high performance on a larger dataset but in other datasets; these models will be vulnerable to overfitting even though their performance is. On the other hand proposed hybrid model [11] Conv-HAN shows high performance just on 2 datasets, not as mentioned in [11]. Naive Bayes is a good choice which definitely attracts attention for future exploration with a larger dataset.

7 Conclusion and Future Work

In this study, we present an overall performance analysis of different approaches on four different datasets.We show that Naive Bayes with n-gram can achieve a result analogous to models based on a neural network. Our results show that after a lot of preprocessing of a relatively small data set. Our future plan is to experiment with our application code on a larger dataset to find out how the traditional model like Naive Bayes competes with deep learning models to detect fake news. Thus the modification of models based on deep learning in order to have more relevant results.

References

1. Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. ACM SIGKDD Explorations Newsletter, 19(1):22–36, 2017.
2. William Yang Wang. " liar, liar pants on fire": A new benchmark dataset for fake news detection. arXiv preprint arXiv:1705.00648, 2017.
3. Julio CS Reis, André Correia, Fabrıcio Murai, Adriano Veloso, Fabrıcio Benevenuto, and Erik Cambria. Supervised learning for fake news detection. IEEE Intelligent Systems, 34(2):76–81, 2019.
4. Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In Proceedings of the 2017 ACM on Conference
5. Eugenio Tacchini, Gabriele Ballarin, Marco L. Della Vedova, Stefano Moret, and Luca de Alfaro. Some like it hoax: Automated fake news detection in social networks.
6. David R. Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 24, pages 1953–1961. Curran Associates, Inc., 2011
7. Ahmed, H., Traore, I., Saad, S.: Detecting opinion spams and fake news using text classification. Secur. Priv. 1(1) (2017). https://onlinelibrary.wiley.com/doi/full/10.1002/ spy2.9
8. Pérez-Rosas, V., Kleinberg, B., Lefevre, A.,Mihalcea, R.: Automatic detection of fake news.In: Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA, 20–26 August, pp. 3391–3401 (2018)
9. Bajaj, S.: The Pope Has a New Baby! Fake News Detection Using Deep Learning.
10. Arvinder Pal Singh Bali(), Mexson Fernandes, Sourabh Choubey,and Mahima Goel :Comparative Performance of Machine Learning Algorithms for Fake News Detection.In : Third International Conference Advances in Computing and Data Sciences, ICACDS 2019 Ghaziabad, India, April 12–13, 2019
11. Khan, Junaed Younus et al. (2019). “A Benchmark Study on Machine Learning Methods for Fake News Detection”. In: CoRR abs/1905.04749.
12. Yang Yang, Lei Zheng, Jiawei Zhang, Qingcai Cui, Zhoujun Li, and Philip S. Yu. Ti-cnn: Convolutional neural networks for fake news detection
13. John Houvardas and Efstathios Stamatatos. N-gram feature selection for authorship identification. Artificial Intelligence: Methodology, Systems, and Applications, pages 77–86, 2006.
14. Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. A stylometric inquiry into hyperpartisan and fake news. arXiv preprint arXiv:1702.05638, 2017
15. Sadia Afroz, Michael Brennan, and Rachel Greenstadt. Detecting hoaxes, frauds, and deception in writing style online. In ISSP’12.
16. Kai Shu, Suhang Wang, Jiliang Tang, Reza Zafarani, and Huan Liu. User identity linkage across online social networks: A review. ACM SIGKDD Explorations Newsletter, 18(2):5–17, 2017
17. Ashutosh Garg and Dan Roth. Understanding probabilistic classifiers. ECML’01.
18. Hanselowski, Andreas et al. (2018). “A Retrospective Analysis of the Fake News Challenge Stance Detection Task”. In: CoRR abs/1806.05180.
19. Shlok Gilda. Evaluating machine learning algorithms for fake news detection. In Research and Development (SCOReD), 2017 IEEE 15th Student Conference on, pages 110–115. IEEE, 2017
20. Sepp Hochreiter and J ̈urgen Schmidhuber. Long short-term memory. Neural Computation, 9:1735–1780, 1997
21. Kaliyar RK, Goswami A, Narang P, Sinha S, 2020. FNDNet – a deep convolutional neural network for fake news detection. Cogn Syst Res 61: 32–44.
22. Ahmed, H., Traore, I., Saad, S.: Detecting opinion spams and fake news using text classification. Secur. Priv. 1(1) (2017). https://onlinelibrary.wiley.com/doi/full/10.1002/ spy2.9
23. W. Y. Wang,”liar, liar pants on fire”: A new benchmark dataset for fake news detection,” arXiv preprint arXiv:1705.00648, 2017
24. Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 1532–1543, 2014
Understanding-LSTMs