JUSTDeep at NLP4IF 2019 Shared Task: Propaganda Detection using Ensemble Deep Learning Models

Hani Al-Omari
Jordan University of Science and Technology
Computer Science Department
Irbid, Jordan
alomarihani1997@gmail.com

Malak Abdullah
Jordan University of Science and Technology
Computer Science Department
Irbid, Jordan
mabdullah@just.edu.jo

Ola Al-Titi
Jordan University of Science and Technology
Computer Science Department
Irbid, Jordan
oaaltiti18@cit.just.edu.jo

Samira Shaikh
University of North Carolina at Charlotte
Computer Science Department
NC, USA
samirashaikh@uncc.edu

Abstract

The internet and the high use of social media have enabled the modern-day journalism to publish, share and spread news that is difficult to distinguish if it is true or fake. Defining "fake news" is not well established yet, however, it can be categorized under several labels: false, biased, or framed to mislead the readers that are characterized as propaganda. Digital content production technologies with logical fallacies and emotional language can be used as propaganda techniques to gain more readers or mislead the audience. Recently, several researchers have proposed deep learning (DL) models to address this issue. This research paper provides an ensemble deep learning model using BiLSTM, XGBoost, and BERT to detect propaganda. The proposed model has been applied on the dataset provided by the challenge NLP4IF 2019, Task 1 Sentence Level Classification (SLC) and it shows a significant performance over the baseline model.

1 Introduction

The spread of news has been transformed from traditional news distributors to social media feeds. However, content on social media is not properly monitored (Granik and Mesyura, 2017). It is difficult to distinguish trusted, credible news from untrustworthy news. This has raised questions about the quality of journalism and enabled the term "fake news". Identifying an article as fake news relies on the degree of falsity and intentionality of spreading the news. There are various types of fake or misleading news, such as publishing inaccurate news to reach a wide audience, publishing untruths with the intention to harm a person or organization, or publishing false news without checking all the facts. News with propaganda are called Propagandistic news articles, that are intentionally spread to mislead readers and influence their minds with a certain idea, for political, ideological, or business motivations (Tandoc Jr et al., 2018; Brennen, 2017).

Detecting fake news and propaganda is getting more attention recently (Jain and Kasbe, 2018; Helmstetter and Paulheim, 2018; Bourgonje et al., 2017), however, the limited resources and corpora is considered the biggest challenge for researchers in this field. In this work, we use the corpus provided by the shared task on fine-grained propaganda detection (NLP4IF 2019) (Da San Martino et al., 2019). The corpus consists of news articles in which the sentences are labeled as propagandistic or not. The goal of the challenge is to build automatic tools to detect propaganda. Knowing that deep learning is outperforming traditional machine learning techniques, we have proposed an ensemble deep learning model using BiLSTM, XGBoost, and BERT to address this challenge. Our proposed model shows a significant performance F1-score (0.6112) over the baseline model (0.4347). The key novelty of our work is using word embeddings and a unique set of semantic features, in a fully connected neural network architecture to determine the existence of propagandistic news in the article.

The remainder of this paper is organized as follows. Section 2 gives a brief description of the existing work in detecting fake news and propaganda. Section 3 provides a dataset description and the extracted features. Section 4 proposes the system architecture to determine the presence of propaganda in an article. Section 5 presents the evaluations and results. Finally, section 6 concludes with future directions for this research.

2 Related Work

Fake news and propaganda are hard challenges that face society and individuals. Detecting fake

Proceedings of the 2nd Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda, pages 113–118
Hong Kong, China, November 4, 2019

news and propaganda is increasingly motivating researchers (Jain and Kasbe, 2018; Helmstetter and Paulheim, 2018; Aphiwongsophon and Chongstitvatana, 2018; Barrón-Cedeño et al., 2019; Orlov and Litvak, 2018). The researchers in Jain and Kasbe (2018) proposed an approach for fake news detection using Naive Bayes classifier, where they applied the model on Facebook posts. The dataset was produced by GitHub that contains 6335 training samples. The results showed that using Naive Bayes classifier with n-gram is better than not using n-gram. Gilda (2017) explored Support Vector Machines, Stochastic Gradient Descent, Gradient Boosting, Bounded Decision Trees, and Random Forests to detect fake news. Their dataset was acquired from signal media and a list of sources from OpenSources.co, to predict whether the articles are truthful or fake.

In Helmstetter and Paulheim (2018), the researchers modeled the fake news problem as a two-class classification problem and their approach was a fake news detection system for Twitter using a weakly supervised approach. Naive Bayes, Decision Trees, Support Vector Machines (SVM), and Neural Networks had been used as basic classifiers with two ensemble methods, Random Forest and XG Boost, using parameter optimization on all of those approaches. In addition, the researchers in (Aphiwongsophon and Chongstitvatana, 2018) proposed a fake news detection model using Naive Bayes, Neural Network and SVM. The dataset collected by their team using TwitterAPI for a specified period between October 2017 to November 2017. The authors in (Bourgonje et al., 2017; Chaudhry et al., 2017) provided a platform to detect the stance of article titles based on their content on Fake News Challenge (FNC-1) dataset[1].

For identifying propagandistic news articles and reducing the impact of propaganda to the audience, (Barrón-Cedeño et al., 2019) provided the rst publicly available propaganda detection system called proppy, which is a real-world and real-time monitoring system to unmask propagandistic articles in online news. The system consists of four modules, which are article retrieval, event identication, deduplication and propaganda index computation. Moreover, (Gavrilenko et al., 2019) applied several neural network architectures such as Long Short-Term Memory(LSTM), hierarchical bidirectional LSTM (H-LSTM) and Convolutional Neural Network (CNN) in order to classify the text into propaganda and non-propaganda. They have used different word representation models including word2vec, GloVe and TF-IDF (Pennington et al., 2014; Mikolov et al., 2013). The results showed that CNN with word2vec representation outperforms other models with accuracy equal to 88.2%. (Orlov and Litvak, 2018) provided an unsupervised approach for automatic identification of propagandists on Twitter using behavioral and text analysis of users accounts. Their proposed approach was applied on dataset that was retrieved from Twitter and collected using the Twitter stream API. Seven suspicious accounts were detected by the approach and it achieved 100% precision.

In contrast to these prior works reviewed, our work is different as we have investigated several Neural Network approaches to determine the most appropriate model for detecting propagandistic sentences in news article. We test the hypothesis that propagandistic news articles would contain emotional and affective words to a greater extent than other news articles.

3 Dataset and Extracted Features

The provided dataset for the NLP4IF 2019 Task 1 is described in (Da San Martino et al., 2019). The corpus consists of 350 articles for training and 61 articles for development for a total of 411 articles in plain text format. The title is followed by an empty row and the content of the article starting from the next row, one sentence per line. There are 16975 sentences in the training data, where 12244 are non-propaganda and 4721 are propaganda.

3.1 Data preprocessing

In our model, text preprocessing has been performed for each sentence of training and development set that includes: removing punctuation, cleaning text from special symbols, removing stop words, clean contractions, and correct some misspelled words.

3.2 Features

In our approach, we have 449 dimensions for our extracted features that are obtained as the following: Each line of text is represented as a 300-dimensional vector using the pretrained Glove embedding model (Pennington et al., 2014).

[1] http://www.fakenewschallenge.org

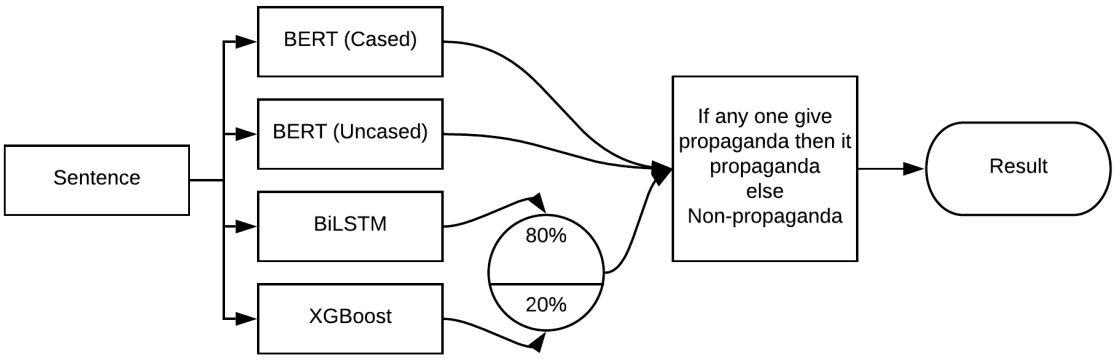

Figure 1: The architecture of our approach

It is worth mentioning that we have also experimented word2vec embedding model that is trained on Google News (Mikolov et al., 2013) but the results were not promising. Our hypothesis is that emotional and affective words will characterize fake news more strongly than neutral words. Accordingly, each line of text is represented as 149-dimensional vector by concatenating three vectors obtained from AffectiveTweets Weka-package (Mohammad and Bravo-Marquez, 2017; Bravo-Marquez et al., 2014), 43 features were extracted using the lexical resources; two-dimensional vector using the sentiments strength feature from the same package, and the final 100-dimensional vector is obtained by vectorizing the text into embeddings (c.f. Table 1).

Features	dimension
Glove	300
TweetToEmbeddings	100
TweetToInputLeixicon	4
TweetToLexicon	43
TweetToSentiStrength	2

Table 1: Features used in our approach

4 Our Approach

The architecture of our system consists of four sub-models: BiLSTM sub-model, XGBoost sub-model, BERT Cased and UnCased model (Figure 1). The description of these sub-models are in the following subsections, we have combined the Cased and UnCased Bert model in one subsection.

4.1 BiLSTM

In this sub-model, we have used use the Bidirectional Long Short-Term Memory (BiLSTM) (Schuster and Paliwal, 1997). The architecture of this sub-model as shown in Figure 2. There are two inputs that are feeding two different network architectures.

The first input is the encoded sentence to embedding layers, which is a lookup table that consists of 300-dimensional pretrained Glove vector to represent each word. This input goes into two BiLSTM layers each with 256 nodes and 0.2 dropout to avoid overfitting. Then, the output from BiLSTM layer is concatenated with Global Max Pooling and Global Average Pooling.

The second input is extracted using AffectiveTweets package as described earlier. The 145-dimensional vector feeds a fully connected neural network with four dense hidden layers of 512, 256, 128, 64 neurons, respectively. We found that the best activation function is ReLU (Goodfellow et al., 2013). A dropout of 0.2 has been added to avoid overfitting. After that we feed it into the previous concatenation layer. A fully connected neural network with four dense hidden layers of 512, 256, 128, 64 neurons for each layer has been applied after the concatenation layer. The activation function for each layer is ReLU, and between them there is a 0.2 dropout.

The output layer consists of 1-sigmoid neuron to predict the class of the sentence. For optimization, we have used Adam optimizer (Kingma and Ba, 2014) with 0.0001 learning rate and binary cross-entropy as a loss function. We have saved the output prediction weights to predict the testing datasets. The fit function uses number of epochs=

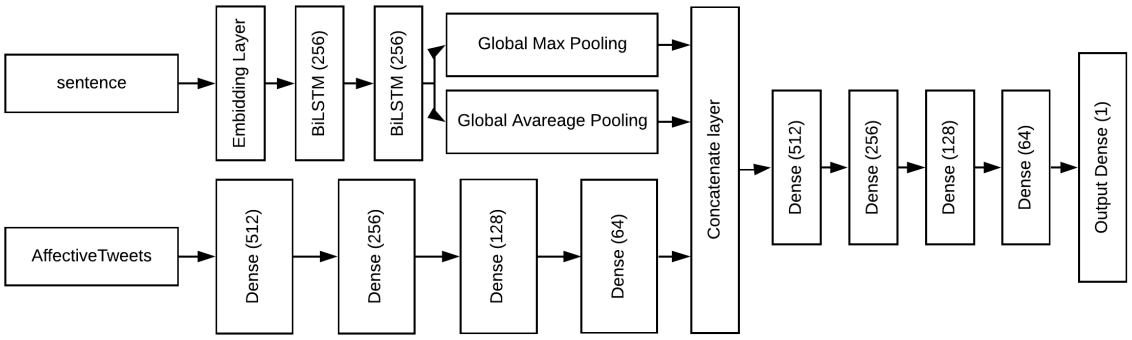

Figure 2: The architecture of BiLSTM sub-model

Features	StopWord	Cased	F1	Precision	Recall
Glove + AffectiveTweets	With	Yes	0.564600	0.630000	0.511502
Glove + AffectiveTweets	With	No	0.550273	0.648897	0.477673

Table 2: BiLSTM result on development data set

100, batch size= 512, validation split= 33% (See Table 2).

4.2 XGBoost

XGBoost (Chen and Guestrin, 2016) is a decision-tree ensemble machine learning algorithm that uses gradient boosting framework. It relies on an iterative method where new models are trained to correct previous model errors. Moreover, it is an optimized implementation of Gradient Boosting Decision Tree (GBDT) that provides a highly-efficient and parallel tree boosting. XGBoost has many hyperparameters that need tweaking. So, we have used Grid search to find the best values for the parameters. Also, we have chosen binary logistic as there are only two classes. Table 3 summarizes XGBoost hyperparameters. It is worth mentioning that we have handled the word embedding by summing words vectors in one sentence and feed it into XGBoost, see Table 4.

Hyperparameter	Value
Number of trees (n estimators)	1200
Learning Rate	0.1
Max Depth	3
Objective	binary:logistic
gamma	0.5
subsample	0.8

Table 3: XGBoost Hyperparameter

4.3 BERT

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) is considered a new pretrained representations which obtains state-of-the-art results on wide variety of natural language processing tasks. BERT has many hyperparameters that need tweaking and after several experiments we adjust the best values for our model. There are two types of pretrained models, BERT-Base and BERT Large (we adopted the base model as it needs less memory). In each type, there are 5 pretrained models, however, we have used Uncased, Cased and Multilingual-Cased. We have noticed that using Uncased and Cased models with ensembling between them gives the best results (Table 5).

5 Results and Evaluation

One of the key findings is noticing that BERT model gives better prediction than the other models, which indicates that BERT can understand the text better than the other models.

In our experiments, we tried several combinations between sub-models. Using the predictions from the BiLSTM and XGBoost models for the development and test datasets, we noticed that the best results are performed with giving BiLSTM sub-model a weight of 0.8 and XGBoost sub-model a weight of 0.2. Combining both results with argmax the predictions to produce a partial result. Regarding the BERT cased and Uncased

Features	StopWord	Cased	F1	Precision	Recall
Glove (Common Crawl)	With	Yes	0.501667	0.652928	0.407307
Glove (Wiki-300)	With	No	0.498328	0.652079	0.403248
Glove+AffectiveTweets (Common Crawl)	With	Yes	0.479932	0.650463	0.380244
Glove+AffectiveTweets (Wiki-300)	With	No	0.480269	0.632743	0.387009

Table 4: XGBoost results on development dataset

Type	seq length	batch size	lr	epochs	StopWord	F1	Precision	Recall
Cased	400	4	1e-5	3	With	0.590288	0.671848	0.526387
Cased	**150**	**8**	**1e-5**	**3**	**With**	**0.600304**	**0.684575**	**0.534506**
Cased	150	8	1e-5	3	Without	0.563694	0.684720	0.479026
Uncased	**400**	**4**	**1e-5**	**3**	**With**	**0.622781**	**0.686786**	**0.569689**
Uncased	150	4	1e-5	3	With	0.573405	0.663701	0.504736
Uncased	150	4	1e-5	3	Without	0.570533	0.677840	0.492558

Table 5: BERT result on development dataset

	F1	Precision	Recall
BERT (Cased) + BERT (Uncased)	0.654671	0.669972	0.640054
BERT (Cased) + BERT (Uncased) + BiLSTM	0.665897	0.580483	0.780785
BERT (Cased) + BERT (Uncased) + BiLSTM (.8) + XGBoost (.2)	**0.674534**	**0.623421**	**0.734777**
BERT (Uncased) + BiLSTM (.5) + XGBoost (.5)	0.641975	0.650904	0.633288
BERT (Uncased) + BiLSTM (.8) + XGBoost (.2)	0.646542	0.543860	0.797023
BERT (Uncased) + BiLSTM	0.633787	0.545366	0.756428

Table 6: Ensembling result on development dataset

result, we have combined both of them together by checking if the 4 models predict that the sentence is non-propaganda then it will be labeled as non-propaganda, otherwise it will be labeled as Propaganda. Table 6 illustrates the best F1 score on the prediction.

6 Conclusion

In this paper, we have investigated several models and techniques to detect if a sentence in an article is propaganda or not. Experimental results showed that the ensemble of using BiLSTM, XGBoost, and BERT has achieved the best results. Also, the process of analyzing and extracting features, such as AffectiveTweets, has a major role in improving the BiLSTM model. The evaluations are performed using the dataset provided by NLP4IF Shared task. The proposed model has been ranked the seventh place among 26 teams. The F1-score that is achieved by our model is 0.6112 which outperformed the baseline model (0.4347) and it is (0.02) away from the first team. We strongly believe that the use of affectivetweets and the lexical features serve well to distinguish between propaganda vs. non-propaganda news.

References

Supanya Aphiwongsophon and Prabhas Chongstitvatana. 2018. Detecting fake news with machine learning method. In *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 528–531. IEEE.

Alberto Barrón-Cedeño, Giovanni Da San Martino, Israa Jaradat, and Preslav Nakov. 2019. Proppy: A system to unmask propaganda in online news. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9847–9848.

Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. 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*, pages 84–89.

Felipe Bravo-Marquez, Marcelo Mendoza, and Barbara Poblete. 2014. Meta-level sentiment models for

big social data analysis. *Knowledge-Based Systems*, 69:86–99.

Bonnie Brennen. 2017. Making sense of lies, deceptive propaganda, and fake news. *Journal of Media Ethics*, 32(3):179–181.

Ali K Chaudhry, Darren Baker, and Philipp Thun-Hohenstein. 2017. Stance detection for the fake news challenge: identifying textual relationships with deep neural nets. *CS224n: Natural Language Processing with Deep Learning*.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM.

Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. 2019. Fine-grained analysis of propaganda in news articles. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, EMNLP-IJCNLP 2019, Hong Kong, China.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Olena Gavrilenko, Yurii Oliinyk, and Hanna Khanko. 2019. Analysis of propaganda elements detecting algorithms in text data. In *International Conference on Computer Science, Engineering and Education Applications*, pages 438–447. Springer.

Shlok Gilda. 2017. Evaluating machine learning algorithms for fake news detection. In *2017 IEEE 15th Student Conference on Research and Development (SCOReD)*, pages 110–115. IEEE.

Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. 2013. Maxout networks. *arXiv preprint arXiv:1302.4389*.

Mykhailo Granik and Volodymyr Mesyura. 2017. Fake news detection using naive bayes classifier. In *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, pages 900–903. IEEE.

Stefan Helmstetter and Heiko Paulheim. 2018. Weakly supervised learning for fake news detection on twitter. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 274–277. IEEE.

Akshay Jain and Amey Kasbe. 2018. Fake news detection. In *2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, pages 1–5. IEEE.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Saif M Mohammad and Felipe Bravo-Marquez. 2017. Wassa-2017 shared task on emotion intensity. *arXiv preprint arXiv:1708.03700*.

Michael Orlov and Marina Litvak. 2018. Using behavior and text analysis to detect propagandists and misinformers on twitter. In *Annual International Symposium on Information Management and Big Data*, pages 67–74. Springer.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Edson C Tandoc Jr, Zheng Wei Lim, and Richard Ling. 2018. Defining fake news a typology of scholarly definitions. *Digital journalism*, 6(2):137–153.