Linguistic Cues Unveiling Fake News

Tianzong Wang
tianzong.wang@epfl.ch

Orazio Rillo
orazio.rillo@epfl.ch

Abstract

The purpose of our project is to come up with a solution that can be utilized by users to detect and filter out misleading news articles based on their content only. Inspired by (Niculae et al., 2015), where linguistic cues are investigated and proved to presage signals in changes of inter-personal relationships, or more explicitly, betrayal detection, we extended the method to another application domain: fake news detection. Some conversational features were replaced with other features that cope with written contents. The code is available under this[1] repository.

1 Introduction

The recent advent of social media has paved the path for a large-scale, low entry-level, faster and more convenient information dissemination. Some sources, such as news articles, play an important role in our daily decision-making. A commonly accepted definition of fake news is: fictitious articles deliberately fabricated to deceive readers (Aldwairi and Alwahedi, 2018). With the use of deceptive words, users can easily get infected by these online fake news. For instance, after the 2016 US president election, a post-election report showed that online social networks account for more than 41.8% of the fake news data traffic in the election (Allcott and Gentzkow, 2017). It is thus important for us to efficiently detect fake news from authentic sources, to improve the trustfulness of online information.

2 Related Work

Previous works in fake news detection focus on either knowledge, style-based linguistic features or stance, social based contextual features. Fake news pieces are intentionally created for financial or political gain rather than to report objective claims, they often contain opinionated and inflammatory language (Shu et al., 2017). It is thus reasonable to exploit the linguistic difference between fake and authentic news. Afroz et al. showed that imitation and obfuscation can be captured from syntactical, lexical and grammar features. Kwon et al. showed temporal, structural and other linguistic features can be used for online rumor detection with a prominent performance. Social context based methods address relevant users or content creators' social engagements to capture the auxiliary information. For example, Tacchini et al. constructed a bipartite network of user and Facebook posts using the "like" stance information, to classify Facebook posts as hoaxes or non-hoaxes with high accuracy. Recent methods incorporated with the use of deep neural networks to further improve the performance. For example, Agarwal et al. used convolutional and recurrent neural networks, as well as GloVe (Pennington et al., 2014) word embeddings for feature extraction and fake news prediction.

3 Dataset

We used the fake news detection dataset from Kaggle [2]. It consists of a total 3988 news article pieces, among which 1872 are labeled as fake. The dataset contains four attributes: URLs, news headline, body as well as the label. Although URLs can be an important attribute on this task, since contents published on a more popular source are less

[1] https://github.com/epfl-ada/ada-2020-project-milestone-p3-p3_thot

[2] https://www.kaggle.com/jruvika/fake-news-detection?select=data.h5

likely to be misleading, we decided to discard this feature and to focus on linguistic ones only.

4 Methods

4.1 Feature Extraction

Sentiment. News articles should theoretically be a neutral reveal of facts without reflecting subjective sentimental responses (Schramm, 1949). The purpose of fake news is to mislead rather than state the truth, thus it is likely for them to embed some sentiment. We provided two methods to retrieve sentiment score. Firstly, we use SentimentIntensityAnalyzer from NLTK library, and calculate the sentiment score by tokenizing each sentence and taking the average compounded sentiment score of all sentences. We also fine-tuned a BERT (Devlin et al., 2019) based sentence-level sentiment classifier on the Google Play Store Apps dataset [3]. The final sentiment score is calculated by taking the difference of positive and negative sentences, divided by the total number of sentences from an article.

Subjectivity. Similarly, fake news are expected to expose more subjectivity. We extracted the subjectivity score with TextBlob, a simple API to access general methods on basic NLP tasks.

Talktiveness. Another aspect to consider is talktiveness. A fact-stating news article, compared with a misleading, pervasive piece, is likely to be more concise in its expression. We measure talktiveness by calculating the average number of sentences and the average number of words per sentence.

Discourse. We calculated the average number of explicit discourse words per sentence by first parsing each sentence using the Stanford Parse Tree[4] parser and then using a script that takes inspiration from an online resource that we will provide in footnote [5].

Planning. We expect that fake news will use more planning markers to influence the future decisions of readers. To capture planning, we collected a list of future tense verbs and then calculated the average number of planning these verbs per sentence for each article.

Dictionary Percentage. An actual article is usually reviewed multiple times before being published. A fake one, on the other hand, is more inclined to contain grammatical mistakes. Because of this reason we decided to introduce this original feature, which is obtained by computing for each article the percentage of words that exist on the English dictionary.

4.2 Model Setup

For the classification task, we first used a logistic regression model with 5-fold cross validation, to keep consistency from the original paper. We also fitted a SVM classifier with rbf kernels, to investigate the impact of model complexity on our task.

5 Results and Discussion

5.1 Exploratory Data Analysis

Firstly, by looking at the correlation heat map plot in Figure 1, we did not notice any highly correlated pair of features except for (number of words, number of sentences) and (average number of discourse markers, average sentence length). The first one is always trivially true. The second pair's correlation is also not difficult to see: indeed the longer is a sentence, the more elaborated it is and thus requires discourse markers to link its subsentences. However, we still decided to include all the extracted features in our model, since there was not a huge redundancy and the model is additionally very simple.

Among those, we selected two features in particular to be shown in Figure 2: (1) the percentage of words of the articles present in the dictionary and (2) the subjectivity score of the articles. The percentage of words present in the dictionary is a particularly interesting feature since it is consistently very high in real articles, while it is not in fake ones. So an high number of grammatical mistakes or dialectal words in an article would immediately unveil the fact that it is a fake one. On the contrary, even thought the subjectivity used by fake news writers is usually higher than the one used in real articles (since professional journalists studied and trained to narrate facts in the most objective way possible) it cannot be used as the only feature used for fake news prediction. Its noise, indeed, can easily lead to a mistake.

[3] https://www.kaggle.com/lava18/google-play-store-apps?select=googleplaystore_user_reviews.csv

[4] https://nlp.stanford.edu/nlp/javadoc/javanlp-3.5.0/edu/stanford/nlp/trees/Tree.html

[5] https://github.com/erzaliator/DiscourseMarker

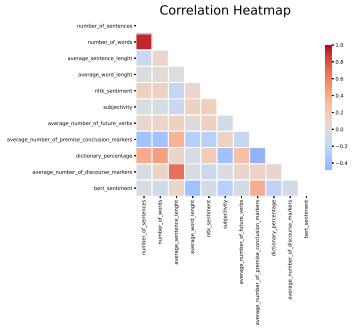

Figure 1: Correlation heatmap of the features.

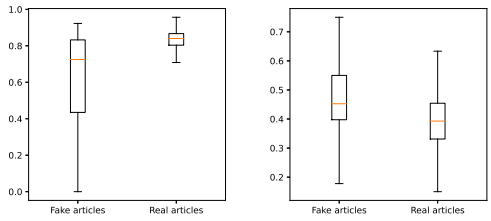

Figure 2: Box plots comparing fake and real articles' distributions of the the percentage of words of the articles in the English dictionary (left) and the subjectivity score of the articles (right).

5.2 Logistic Regression

We obtained an average test accuracy of 77.7% with a standard deviation of 0.013, from our 5-fold cross validation logistic regression model. A model summary is included below. Note that here, we refer to the features by: x_1 - BERT sentiment score, x_2 - number of sentences, x_3 - number of words, x_4 - average sentence length, x_5 - average word length, x_6 - NLTK sentiment score, x_7 - subjectivity score, x_8 - number of future verbs, x_9 - number of premise & conclusion markers, x_{10} - dictionary percentage, x_{11} - number of discourse markers.

```
Optimization terminated successfully.
         Current function value: 0.466972
         Iterations 7
                           Logit Regression Results
==============================================================================
Dep. Variable:                  Label   No. Observations:                 3190
Model:                          Logit   Df Residuals:                     3179
Method:                           MLE   Df Model:                           10
Date:                Fri, 18 Dec 2020   Pseudo R-squ.:                0.3239
Time:                        15:32:07   Log-Likelihood:               -1489.6
converged:                       True   LL-Null:                      -2203.1
Covariance Type:            nonrobust   LLR p-value:                1.486e-300
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
x1            -0.7187      0.068    -10.617      0.000      -0.851      -0.586
x2             1.0771      0.174      6.182      0.000       0.736       1.419
x3            -1.8769      0.240     -7.821      0.000      -2.347      -1.407
x4             0.9343      0.073     12.760      0.000       0.791       1.078
x5             0.4159      0.061      6.857      0.000       0.297       0.535
x6             0.2539      0.067      3.768      0.000       0.122       0.386
x7            -0.7694      0.062    -12.447      0.000      -0.891      -0.648
x8            -0.2133      0.052     -4.130      0.000      -0.314      -0.112
x9            -0.0702      0.065     -1.075      0.282      -0.198       0.058
x10            1.2135      0.088     13.857      0.000       1.042       1.385
x11            0.8252      0.184      4.494      0.000       0.465       1.185
==============================================================================
```

Figure 3: Summarize statistics of Logistic Regression

Judging from the coefficients magnitude, talkativeness, subjectivity, sentiment features and grammar mistake counts are highly predictive in detecting fake news. Other features, like the use of planning and discourse markers, are less significant but yet not to be discarded. Since from the ($P > |z|$ column), we can conclude that all feature used are significant enough (p less than 5%). We also included the ROC curve plot. The concave shape tells that the portion of correctly predicted fake news is always greater than that of mis-classified authentic news. We do not want to avoid fake news by discarding real pieces, this model is thus acceptable.

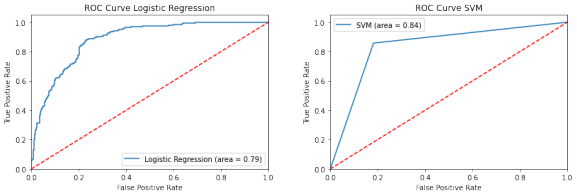

Figure 4: ROC Curves

5.3 SVM

Our SVM model obtained an accuracy score of 83.7%, precision score of 81.5% and recall 85.7%. A more complicated model was indeed able to improve the performance. An ROC curve plot is provided below, in comparison with Logistic Regression Model. Compared with that from logistic regression, a higher AUC value indicates that SVM is a better model choice. An optimal threshold appear at the turning point where the true positive rate is around 80% and false negative rate is 20%.

6 Conclusion

Despite fake news writers' best effort to sound genuine, the authenticity can still be revealed from their writings. Our model, although simple, was able to detect fake news with 80% accuracy. However, feature extraction was not an easy task. For example, our BERT sentiment classifier still takes 4 hours on a single V100 GPU to assign sentiment scores. This rose the time complexity concern. Given the speed of fake content production, our method may not be efficient enough to cope with it on a large scale.

Another concern is that our model can also entail a reverse hacking strategy. People could easily improve the 'authenticity score' of their writings by 'engineering' the features that we individuated e.g. by adjusting the lengths of the sentences or making sure to only use words that are present in an English dictionary, and so end up writing articles that are almost indistinguishable from real ones for linguistic-based filters.

References

Monther Aldwairi and Ali Alwahedi. 2018. Detecting fake news in social media networks. *Procedia Computer Science*, 141:215 – 222. The 9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2018) / The 8th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2018) / Affiliated Workshops.

Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36, May.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Vlad Niculae, Srijan Kumar, Jordan Boyd-Graber, and Cristian Danescu-Niculescu-Mizil. 2015. Linguistic harbingers of betrayal: A case study on an online strategy game.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. volume 14, pages 1532–1543, 01.

Wilbur Schramm. 1949. The nature of news. *Journalism Quarterly*, 26(3):259–269.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective.