COMP6246 MLT REPORT

VERIFYING MULTIMEDIA USE

Sharang Gupta
32196946

01.01.2021

INTRODUCTION

Fake news detection is a critical yet challenging problem in Natural Language Processing (NLP). The rapid rise of social networking platforms has not only yielded a vast increase in information accessibility but has also accelerated the spread of fake news. Given the massive amount of Web content, automatic fake news detection is a practical NLP problem useful to all online content providers, in order to reduce the human time and effort to detect and prevent the spread of fake news.

DATA ANALYSIS

DATASET CHARACTERIZATION

The MediaEval 2015 "verifying multimedia use" dataset consists of social media posts (e.g. Twitter, Facebook and blog posts) for which the social media identifiers are shared along with the post text and some additional characteristics of the post. A set of ground truth labels (i.e. ‘fake’ or ‘real’) are provided in the dataset for both the training and test set.

LABEL BIAS

The data has been divided into a training set containing 14,483 records and a test set containing 3781 records. Both these datasets appear to be skewed with a fake to real ratio of 0.89 and 2.10 respectively. This can be easily fixed with resampling, which was the first step in the process. Upon combining the datasets, the ratio changed to 1.06, a much more balanced distribution.

FEATURE GENERATION

The dataset consists of 7 columns, with fields like tweet id, tweet text, username, user id, image id and timestamp. Tweet id is a randomly generated value with no classifying power, and hence we drop it. Our current analysis does not use the timestamp, and hence we drop it. Username does not affect the tweet, and hence we drop it. As a future scope of our analysis, we could relate certain user and image characteristics to our label, as certain users who regularly post fake news may continue doing so, whereas certain high reputation users would be unlikely to post fake news. Similarly, if a certain image has been used in a fake post, it is likely that the same image used in other posts may indicate that the post is fake. Our analysis in this report will be limited to the tweet text, and the features we can extract from the text itself.

In order to analyse the tweet text, we need to extract text based linguistic features from the tweet, and some of these features are **word count**, **length** of the tweet, **uppercase character count**, **question marks**, **exclamation marks**, **colons**, **mentions**/@, **hashtags**/#, **URL count**, **first order pronoun count**, **second order pronoun count**, **third order pronon count**, **sad and happy emotion count**, as used with a great deal of success in [1](Lamba et al. 5).

Apart from the above mentioned features, we also include the presence of **swear words** as a part of linguistic features and presence of **'via'** as used in [2](Gupta, Aditi et al. 6). We also generate syntactic features like **bag of words(tf-idf)** and **n-grams**, as referenced from [3](Kai et al. 5). Finally, we calculate the text **polarity** and **subjectivity**, as used in [4](Saha et al. 2).

ALGORITHM DESIGN

Earlier research in fake news detection mostly relied on manual feature selection based on psycholinguistic theories of deception and/or computational linguistics, followed by supervised machine learning to build a classifier, but recent NLP researches are now increasingly focusing on the use of new deep learning methods.[5](Girgis et al. 2)

[5](Conroy et al.) Used a hybrid approach of multi-layer linguistic processing, along with network analysis, to obtain an accuracy of 72%, further enhanced by performing cross-corpus analysis of classification models and reducing the size of the input feature vector. This research yielded one of the best results that these features can generate, paving the way for future researchers to explore neural networks.

[7](Samir Bajaj, 2017) used multiple algorithms for neural networks and machine

learning to obtain the highest accuracy, on data obtained from an openKaggle dataset of 13,000 Fake news articles and 50,000 authentic news articles (negative examples for the classifier). He used many techniques from machine learning and neural network such as Logistic Regression, Feedforward Network, RNN (Vanilla, GRU), LSTMs, Bi- LSTMs, CNN with Max-Pooling and CNN with Max-Pooling and Attention, The results proved that GRU gets the better F1 score and best results overall.

[8](Natali Ruchansky et. Al, 2017) proposed a model called CSI, which is built up from deep neural networks and can extract information from different domains and capture temporal dependencies in user engagement with articles, and also select important features, for improving on the previous results. CSI (which is composed of three modules: Capture, Score, and Integrate) evades the cost of manual features selection by incorporating neural networks. The features they use to capture the temporal behavior and textual content in a general way that does not depend on the data context nor requires distributional assumptions. They used two datasets from Twitter and Weibo (real-world social media datasets). CSI gives the best performance overall comparison models and versions.

[9](Huang et al.) uses an ensemble learning model combining four different models called embedding LSTM, depth LSTM, LIWC CNN, and N-gram CNN for fake news detection. In order, to achieve higher accuracy in fake news detection, the optimized weights of the ensemble learning model are determined using the Self-Adaptive Harmony Search (SAHS) algorithm. Experimental methods demonstrated that the proposed model is superior to the state-of-the-art methods, with the highest accuracy of 99.4%

Other Works by [10](William Yang et.al, 2017) presents a new benchmark dataset called LIAR : it's a new, publicly available dataset for fake news detection. He used it with many techniques such as logistic regression, support vector machines, and (Bidirectional-LSTM and CNN) models for deep learning. The results proved that CNN (Convolutional Neural Networks) models are the best.

DATA PREPROCESSING AND FEATURE SELECTION

Data preprocessing involves cleaning our tweets of all stop words, tokenizing it using nltk and performing lemmatization on the tokens. This prepares the text for creating our n-grams(bi-grams, tri-grams) as well as the tf-idf(term frequency-inverse document), which we would use as lexical features.

Upon analysing the features against our label, we can see that some of the features have no meaningful correlation by itself with the output variable. Features like the first and second pronoun do not seem to have the same impact as the third pronoun. Instead of dropping it altogether, combining all the 3 pronoun counts together even beats the third pronoun count, which had the highest impact of the three.

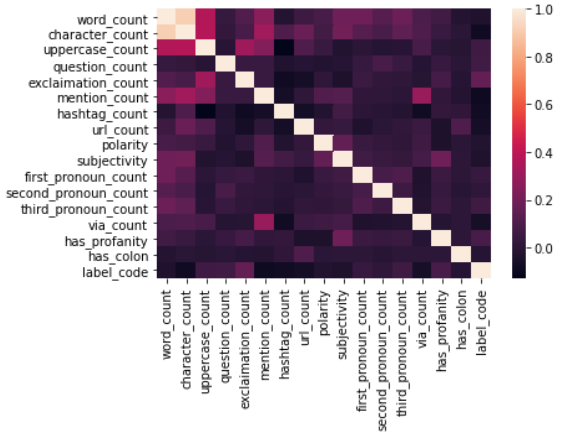


Fig-1 : correlation matrix of all selected features

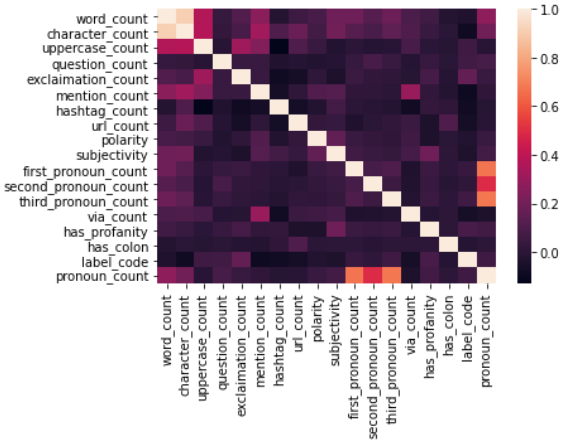


Fig-2 : correlation matrix including pronoun count

Also, subjectivity and polarity do not seem to have as much of an effect on the label as in Fig-1. Combining both of them together seems to yield a better feature, as in Fig-3

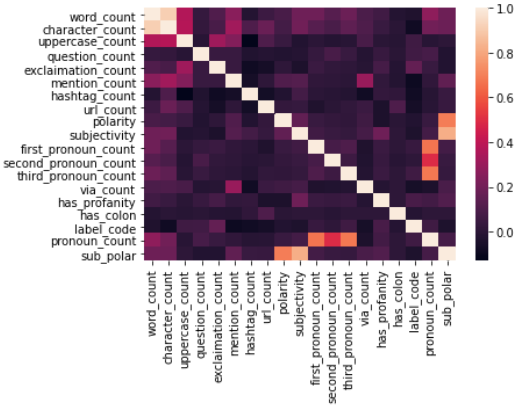


Fig-3 : correlation matrix including sub polar

FEATURE PLOTTING

As we can see, exclamations and questions are much more prominent in fake tweets

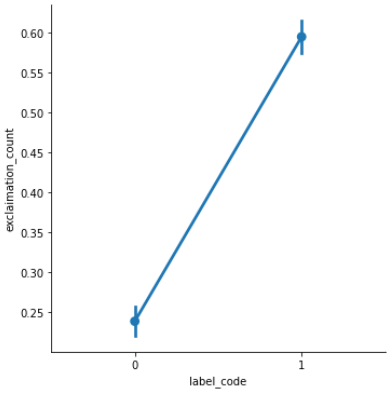


Fig-4 : Exclamations against label

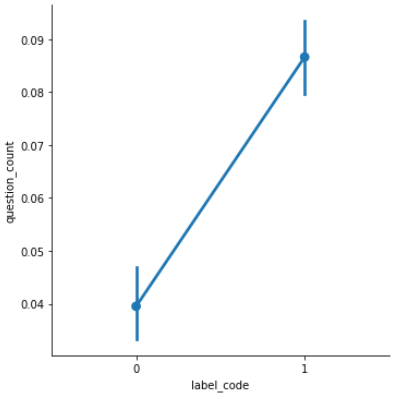
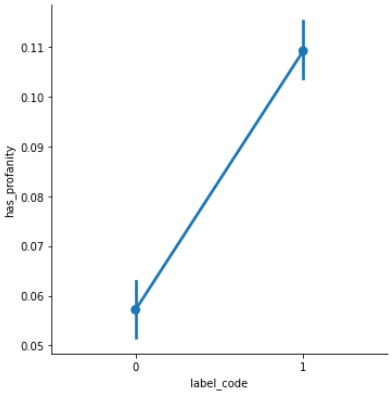
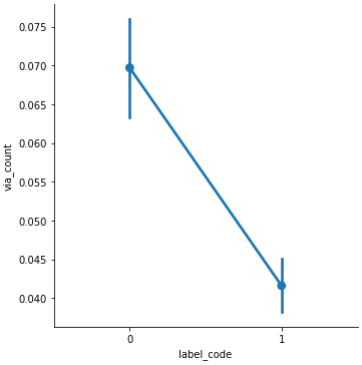


Fig-5 : Questions against label

As we would expect, genuine tweets contain much lesser profanity



A rather surprising find is that the feature ‘via_count’, or the frequency of ‘via’ in the tweet, is a lot more prominent in real tweets as compared to fake ones.



DIMENSIONALITY REDUCTION

As a result of our feature engineering, we can drop the individual features we had combined, i.e, polarity, subjectivity, first_pronoun_count, second_pronoun_count and third_pronoun_count. We have also introduced 2 new features, i.e, prooun_count and sub_polar.

ALGORITHM SELECTION

As backed by the literature review, we can shortlist the following 5 algorithms:

1. Convoluted Neural Networks(CNN)
2. Gated Recurrent Unit(GRU)
3. Capture, Score and Integrate(CSI)
4. Recurrent Neural Network(RNN)
5. Ensemble Learning.

Evaluation

1. Convoluted Neural Networks(CNN) : CNNs are a variant of feed-forward neural networks with a special architecture. The architecture of CNNs usually contains a convolution followed by a pooling operation. Every neuron in a convolutional layer is connected to some region in the input, which is called a local receptive field.[11](Suryani et al.)

   #### ADVANTAGES

   - CNNs are very good feature extractors. This means that you can extract useful attributes from an already trained CNN with its trained weights by feeding your data on each level and tune the CNN a bit for the specific task.
   - Results are more accurate than typical machine learning techniques if tuned better and feeded a good amount of data.
   - CNN weight sharing makes it more efficient in terms of memory and complexity, in a larger scale CNNs would be less complex and save more memory compared to the NN.

   #### DISADVANTAGES

- If the CNN has several layers then the training process takes a lot of time.
- CNN is significantly slower due to an operation known as maxpool.
- CNN requires a large dataset to process and train the neural network.

2. Gated Recurrent Unit(GRU) : Gated recurrent units (GRUs) are a gating mechanism in recurrent neural networks

ADVANTAGES

- GRUs train faster and perform better than LSTMs on less training data.
- GRUs are simpler and thus easier to modify, for example adding new gates in case of additional input to the network. It's just less code in general.
- GRUs address the vanishing gradient problem (values used to update network weights) from which vanilla recurrent neural networks suffer.

DISADVANTAGES

- Slow convergence.
- Low learning efficiency.
- Require lots of data.

3. Capture, Score and Integrate(CSI) : CSI is built up from deep neural networks and can extract information from different domains and capture temporal dependencies in user engagement with articles, and also select important features, for improving on the previous results.

ADVANTAGES

- CSI evades the cost of manual features selection.
- CSI does not depend on the data context, nor requires distributional assumptions.
- CSI yields High performance..

DISADVANTAGES

- Uses deep neural networks, slow to train.
- Low learning efficiency.
- Require lots of data.

4. Recurrent Neural Network(RNN) : A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior. Derived from feedforward neural networks, RNNs can use their internal state (memory) to process variable length sequences of inputs.

ADVANTAGES

- RNN can process inputs of any length.
- Even if the input size is larger, model size does not increase.
- RNN can use their internal memory for processing the arbitrary series of inputs which is not the case with feedforward neural network.
- The weights can be shared across the time steps.

DISADVANTAGES

- Due to its recurrent nature, the computation is slow.
- Training of RNN models can be difficult..
- If we are using relu or tanh as activation functions, it becomes very difficult to process sequences that are very long.
- Prone to problems such as exploding and gradient vanishing.

5. Ensemble Learning : Ensemble learning is the process by which multiple models, such as classifiers or experts, are strategically generated and combined to solve a particular computational intelligence problem

ADVANTAGES

- Bring a decentralized, consensus-based approach to machine learning that helps to refine results and ensure precision.
- Can reduce overfitting and give a better classifier than only a single system.
- Can be 'boosted', which allows one to create a classifier that is tremendously accurate from a set of models that are individually mediocre.

DISADVANTAGES

- Learning time and memory constraints.
- Ensemble methods reduce the model interpretability due to increased complexity.
- Difficult to measure correlation between classifiers from different types of learners.

As we can see from the above comparison, each model has various advantages and disadvantages. All of the above have been used with a high degree of success for identifying fake news as demonstrated in the references. Amongst the models presented above, GRU's are a good choice with regards to memory and learning time constraints, although we have observed better results with others. It all boils down to the trade-off between accuracy of the results versus the resource utilization, and hence, if we were to maximize the accuracy of our results, we should pick the **Ensemble method**. Using this model allows us to leverage multiple models in a combination, and they have proved to

be very effective. For our use case, interpretability is not something we need to worry about, as our end users may not need to be concerned with how it's being classified, but would definitely be concerned if it's classified wrongly. The third place would go to CNNs if we do not mind the extra training time, or could be interchanged with the GRU if that is of concern to us.Hence, an appropriate ranking for the suitability of the algorithms would be as follows :

1. Ensemble Learning
2. CSI model
3. Convoluted Neural Networks
4. Gated Recurrent Unit
5. Recurrent Neural Networks

CONCLUSION

In summary, we can conclude that for a text based classification task like this, we can extract a lot of useful features from within the text itself, and the other features are of not any real use for now. We performed a literature survey on many papers which explore numerous successful algorithms that have been used for fake news detection, and the results have been very promising. After a thorough analysis on all the algorithms, and evaluating their strengths and weaknesses, we have selected an Ensemble Learning model as referenced from [9](Huang et al.), which combines four different models called embedding LSTM, depth LSTM, LIWC CNN, and N-gram CNN for fake news detection.

For further research in this problem, we should explore mining user data along with the tweets and textual features, in order to combine the results from both to obtain a more robust model. The above improvements are suggested because the user characteristics play a role in determining if the content is more or less likely to be fake. A user who tends to post fake news in order to sensationalize and increase his follower count is more likely to post fake content in the future too, whereas a user with a high reputation, say for instance a reputed news channel is less likely to post fake news.

Another alternative is to analyze the images posted along with the tweets, and use machine learning algorithms to detect if the image alongside the post is fake, and this could be a useful feature as the textual content would be linked to the image, and if the image is fake, it's likely that the textual content too is fake.

In conclusion, exploring this dataset and the various references have proven to be

immensely educative, as we had to generate features from text, evaluate them, and then read about multiple algorithms and the vast improvements taking place. We learnt that although individual features by themselves may show a low correlation, combining them can improve the correlation substantially. In the end, we learn of the power and limitless scope of ensemble learning, which allows us to combine multiple models in order to increase accuracy, albeit at the cost of performance. There is still some room for improvement, and we could explore various ensemble combinations in order to further optimise the results.

REFERENCES

1. Gupta, Aditi & Lamba, Hemank & Kumaraguru, Ponnurangam & Joshi, Anupam. (2013). Faking Sandy: characterizing and identifying fake images on Twitter during Hurricane Sandy. WWW 2013 Companion - Proceedings of the 22nd International Conference on World Wide Web. 729-736. 10.1145/2487788.2488033.
2. Gupta, Aditi et al. “TweetCred: Real-Time Credibility Assessment of Content on Twitter.” Lecture Notes in Computer Science (2014): 228–243. Web.
3. Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake News Detection on Social Media: A Data Mining Perspective. SIGKDD Explor. Newsl. 19, 1 (June 2017), 22–36. DOI:https://doi.org/10.1145/3137597.3137600
4. Saha, Shubhodip & Yadav, Jainath & Ranjan, Prabhat. (2017). Proposed Approach for Sarcasm Detection in Twitter. Indian Journal of Science and Technology. 10. 10.17485/ijst/2017/v10i25/114443.
5. S. Girgis, E. Amer and M. Gadallah, "Deep Learning Algorithms for Detecting Fake News in Online Text," 2018 13th International Conference on Computer Engineering and Systems (ICCES), Cairo, Egypt, 2018, pp. 93-97, doi: 10.1109/ICCES.2018.8639198.
6. Conroy, Niall J., Victoria L. Rubin, and Yimin Chen. "Automatic deception detection: Methods for finding fake news." Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact.
7. Bajaj S. “The Pope Has a New Baby! Fake News Detection Using Deep Learning”,2017 .
8. Ruchansky N, Seo S, Liu Y. Csi.” A hybrid deep model for fake news detection”. InProceedings of the 2017 ACM on Conference on Information and Knowledge Management ,2017 .Nov 6. pp. 797-806, ACM.
9. Huang, Yin-Fu & Chen, Po-Hong. (2020). Fake News Detection Using an Ensemble Learning Model Based on Self-adaptive Harmony Search Algorithms. Expert Systems with Applications. 159. 113584. 10.1016/j.eswa.2020.113584.
10. Wang WY. " liar, liar pants on fire": A new benchmark dataset for fake news detection”. arXiv preprint arXiv:1705.00648. 2017, May 1.
11. Dewi Suryani, Patrick Doetsch and Hermann Ney On the Benefits of Convolutional Neural Network Combinations in Offline Handwriting Recognition.