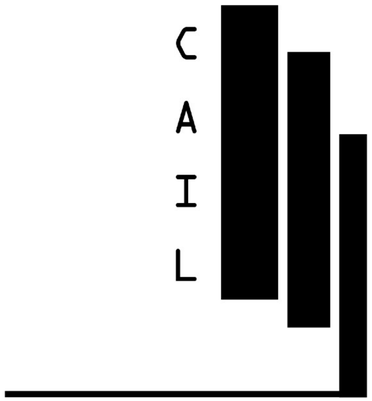
Suggestion of Low-Complexity Fake News Detection AI

Using alphabet frequency data and artificial neural network

HyeonJun Kim (Soongsil University), CELL AI - data science Laboratory, CELL Senior



Preface

Limitations and Focusing Issues & Contact

This research is not fully based on modern natural language processing. Thus, the overall results are not always superior than other algorithms. Rather than reviewing conventional algorithms, this research is focused on minimizing complexity of the algorithms that is needed to process emerging attributes of the system. Also, in this study the dataset in Kaggle (https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset) is used, and the debate whether or not the dataset is real will not be discussed.

CELL AI - data science Laboratory (CAIL) is a undergraduate community of data science and artificial intelligence in Soongsil University, Republic of Korea. CAIL welcomes any comments or feedback on the works that been done by member of CAIL, suggestions of academic support or domestic, international member applications. The author of this study, HyeonJun Kim can be contacted by E-mail.

Formal E-mail: inertia164214@soongsil.ac.kr , Personal E-mail: hyeonjunacademic@gmail.com

Introduction

Introduction

Problem-Suggestive Pilot Study

- Most of former studies focused on the fact that the attribute of higher order than just characters and their order in the text of news has the attribute for fact-checking. For example, linguistic approaches focused on the negative meaning of words itself (Feng & Hirst, 2013), and there have been network approaches as well (Ciampaglia et al., 2015).

- There have been attempts to apply statistical models and algorithms of natural language processing and machine learning like N-gram (Ahmed, Traore & Saad, 2017), and the some of the results are notable (92% accuracy using TF-IDF, Uni-gram, LSVM).

- However, the former studies are using models to evaluate the sequence and the letters of the text or the meaning of the word, the higher order of information complexity. If the analysis needs more data to run, the more processing power, more memory, and more complex algorithms and models are needed as well. Thus, in the perspective of applications for mobile devices or easier and faster processing software applications, the trade of the accuracy and complexity could be reasonable.

Proposed Method

Proposed Method 1-1
Alphabet Probability Vector (APV)

- A complex structure of sentences and words seems impossible to hash into more smaller and representative data size. However, it is well known that the english alphabet has different frequency of use. Therefore, the way to distinguish two different type or purpose of text can be suggested. If the text's meaning depends on the sentences and words that been used, and most of those sentences and words the distinguished by use of different alphabets, it could be possible to hash the uniqueness of the alphabets frequency and the meaning of the whole text. If this hypothesis is right, by subtracting APV of commonly used words and APV of the text that is evaluated, the uniqueness of the frequency becomes more explicit and the algorithm might show more strong capability to accurately find the fake news.

- In this case, exceptions like 'ate' and 'eat' is not considered. It could be suggested that this exceptions make the algorithms performance limited regardless of its form.

- The frequency is divided by the sum of the frequency and stored as a 26 dimensional vector.

- For extracting a vector, a special object is defined and used for decomposing the text and generating the vectors.

Proposed Method 1-2
Alphabet Probability Vector (APV)

- $p_k = \frac{n_k}{N} \quad \left(N = \sum_k n_k \right)$, n_k = (The frequency of k th alphabet in a given text.)

- $v_{apv} = \left[\, p_1 \; p_2 \; \cdots \; p_{26} \, \right]$

- v_0 is a v_{apv} with probabilities of the alphabets of 40,000 words. The data that been used is referred from site run by Department of mathematics, Cornell University (http://pi.math.cornell.edu/~mec/2003-2004/cryptography/subs/frequencies.htmlcc).

- Standard-subtraction model use the modified version of APV which is $v'_{apv} = v_{apv} - v_0$.

Proposed Method 2

ANN with Hidden Layers

- As the language has a multiple stage of emergence, just comparing limited features by simple models cannot extract the pattern of fake news and the trustworthy ones. Thus, artificial neural network is chosen to unpack features of the data. With enough hidden layers, the ANN hopefully distinguish the difference between fake news and trustworthy ones only using the summarized data (in this case, APV of the news text) that contains information (i.e. its trustworthiness) about its original content.

- ANN model is designed by using Keras and Python. As the model is required to find complicated relationship between APV and its original text, initially three hidden layers with 256 dense nodes are used.

Experiments

Experiments

Datasets and Process

- By extracting APV from about 44,000 news text(FT dataset) or news title text(TT dataset) and labeling “fake” (0) or “true” (1), the dataset of 27 columns and about 44,000 rows is obtained. Then the dataset is split into training set (30,000 rows) and evaluation set (14,000+ rows) for training and evaluating ANN.

- Firstly, the initial model of AI is tested.

- For optimizing results, the method of subtracting standard frequency from APV, and changing the dataset is been tested.

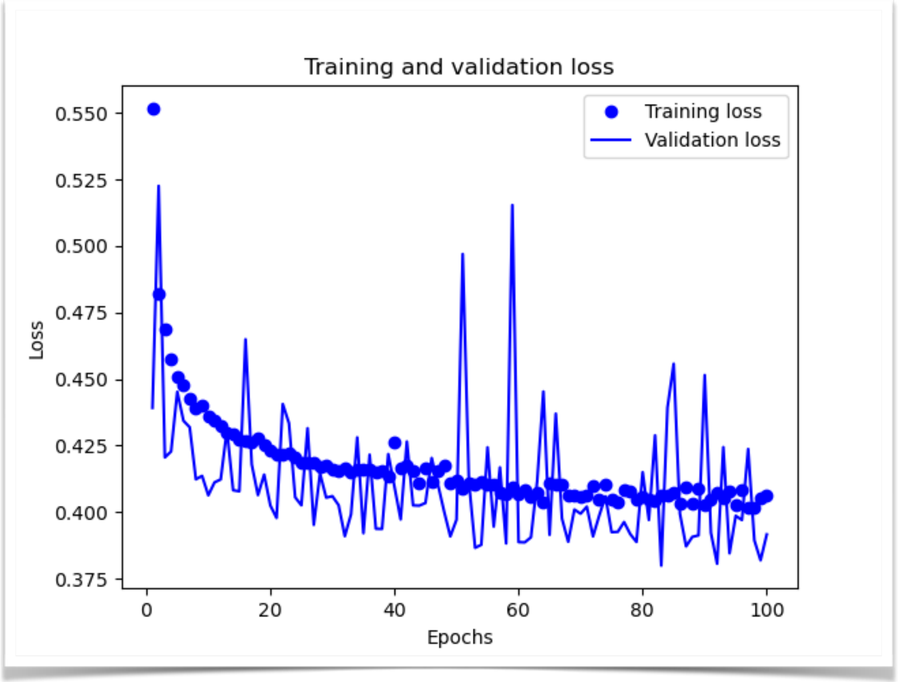
- The training efficiency of the model is evaluated by mean and median of the accuracy of evaluation set during training of ANN. This does not shows the highest performance, but with graphs, this values makes it easier to observe training efficiency at same epochs and batch size.

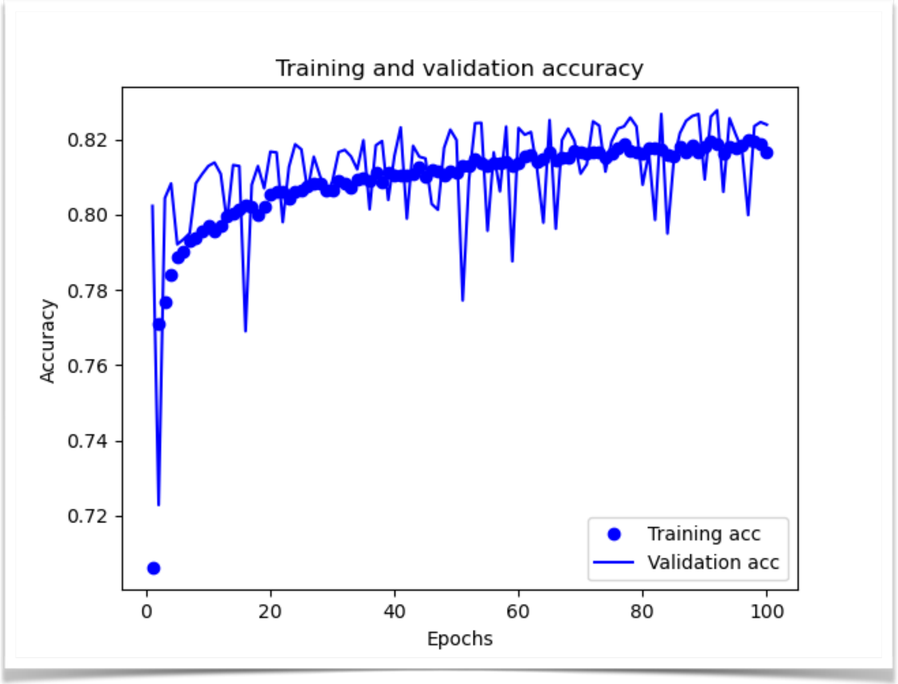
Results

Result 1

Initial Model Evaluation

- One input layer, 3 hidden layers
- APV from FT dataset
- 256 nodes for each layers
- One node output layer using sigmoid function
- Epochs: 100, batch size: 50
- Result (Accuracy):
- Mean: 0.81186 (81.1%)
- Median: 0.81517 (81.5%)

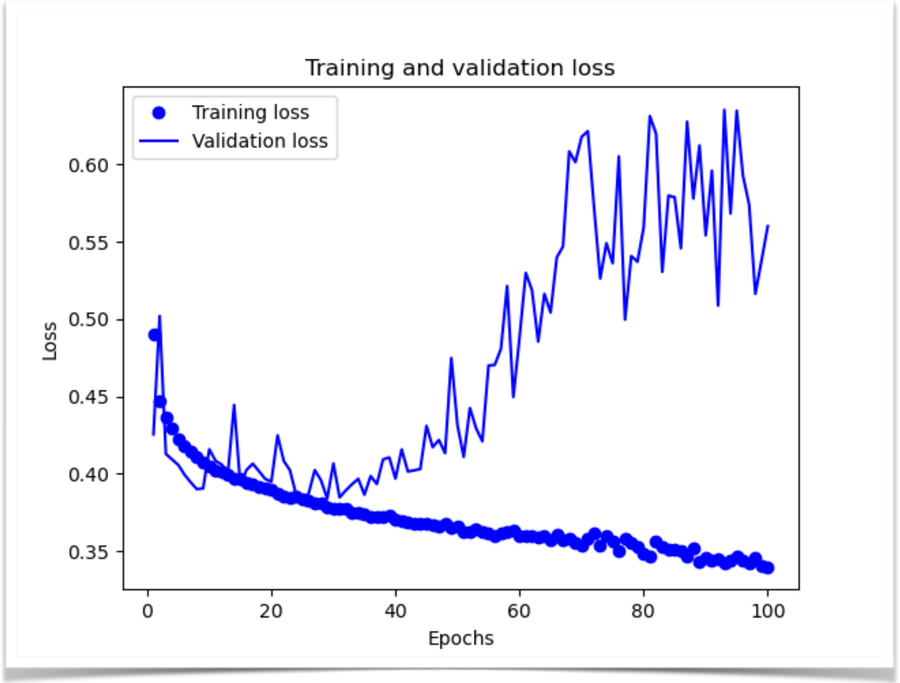


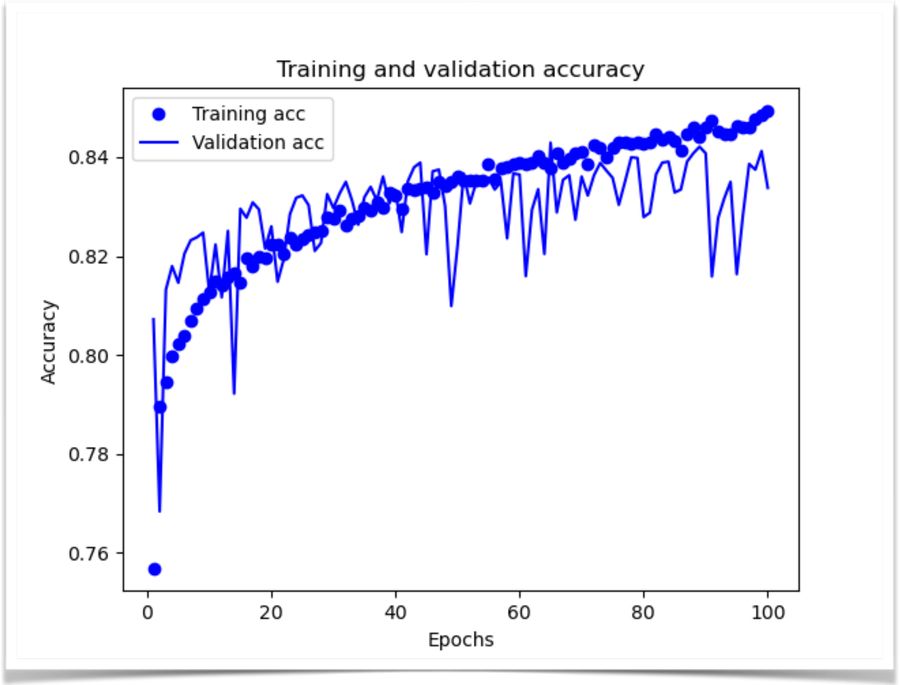


Result 2

Standard-Subtraction Model (SSM)

- One input layer, 3 hidden layers
- APV from FT dataset
- 256 nodes for each layers
- One node output layer using sigmoid function
- Epochs: 100, batch size: 50
- Result (Accuracy):
- Mean: 0.82899 (2.1% improvement)
- Median: 0.83148 (2.0% improvement)
- Adding one frequency number for missing letter actually improved accuracy results.

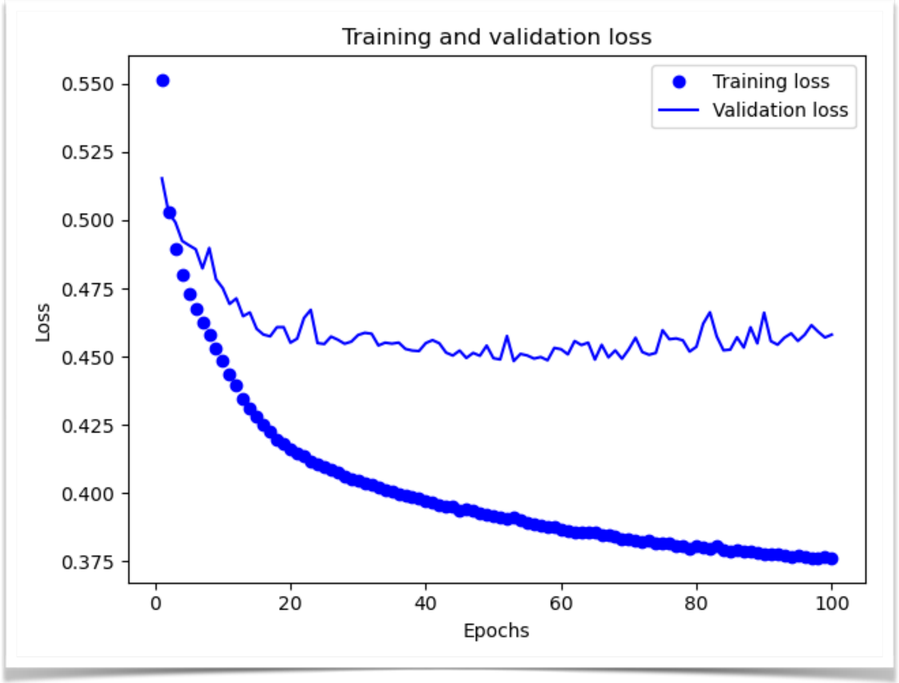


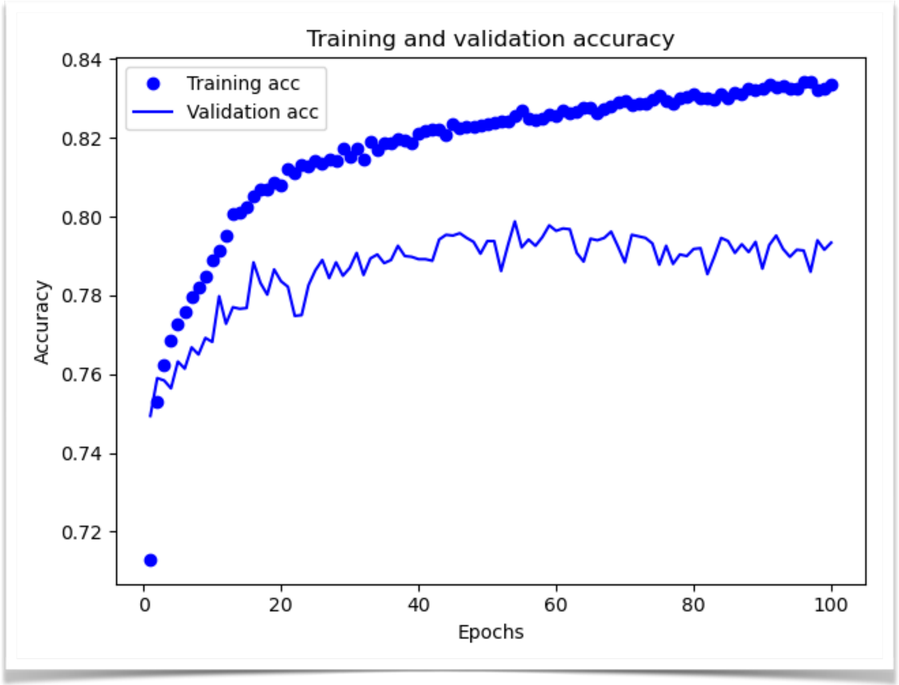


Result 3

News Title Based SSM

- One input layer, 1 hidden layers
- APV from TT dataset
- 32 nodes for each layers
- One node output layer using sigmoid function
- Epochs: 100, batch size: 50
- Result (Accuracy):
- Mean: 0.78702 (78.7%)
- Median: 0.79020 (79%)





Final Results

- For 1000 epochs in 50~200 batch size, the Standard-Subtraction Model base on news text shows 0.8448 (84.4%) accuracy on detecting fake news. (calculated from prediction of 9898 random dataset)

- For 100 epochs in 50 batch size, the Standard-Subtraction Model base on news title text shows 0.7866 (78.6%) accuracy on detecting fake news. (calculated from prediction of 9898 random dataset)

Conclusion

Conclusion 1

- The idea of reducing calculation and memory space is important concept of algorithm analysis. And as internet and its service is going mobile, it is crucial for enhancing time and space efficiency while a program of mobile device is running.

- As the algorithm that is suggested processed only the frequency of alphabets of the whole text, which reduces training dataset into 26 / n (n letter of the original text), this compact training set can make training dataset much more portable in the Internet world. Also, the data is decomposed to the point that just by the APV, it is almost impossible to predict what the original content was, which is good for privacy. And the algorithm still can detect different intention of the text in high accuracy (84.4%).

Conclusion 2

- Also, if chosen correctly, a representative text of the whole text could be used for evaluation, and still have notable accuracy (78.6%), which is about 93% the accuracy of much more complex initial algorithms.

- As the algorithm is simple, compact, and fast processing (ANN with at total 65 nodes), hopefully it can be applied to smartphone applications or mobile browser processing the content without any track of what the content was. If the algorithms is more improved, it seems possible for the algorithms to support security measures like checking whether or not the text-based communication is fake or not, or detecting harmful text-based content on the platform.

- Lastly, it seems that more additional research about the hash-relationship between APV and the original text are needed, to understand why this algorithm work and what are its limitations, and what is more possible using this method.

References

- Conroy, Nadia K., Victoria L. Rubin, and Yimin Chen. "Automatic deception detection: Methods for finding fake news." *Proceedings of the Association for Information Science and Technology* 52.1 (2015): 1-4.

- Ahmed, Hadeer, Issa Traore, and Sherif Saad. "Detection of online fake news using n-gram analysis and machine learning techniques." *International conference on intelligent, secure, and dependable systems in distributed and cloud environments*. Springer, Cham, 2017.