FPT UNIVERSITY

AIP391 Project

Fake News Detection

Thai Duy Bao
FPT University
Ho Chi Minh City, Vietnam
baotdse161680@fpt.edu.vn

Huynh Minh Triet
FPT University
Ho Chi Minh City, Vietnam
triethmse160251@fpt.edu.vn

Abstract

Due to the strong development of the internet in recent years, news all over the world are transmitted faster and faster through news websites, social media, blogs, etc. However, this also make it easier for Fake news to spread, which, sometimes, can lead to the outbreak of mass hysteria and negative impact on society. In this paper, we have we researched and built a machine learning model that can distinguish fake news from real news. Firstly, we introduce two datasets of English and Vietnamese articles. Secondly, we use different algorithms to train our model to predict on each dataset. Finally, we compare the results and draw conclusions. Specifically, we discovered that the English test using TF-IDF Vectorization with Passive-Aggressive Classifier produced the highest accuracy.

I. INTRODUCTION

Along with the development of the 4.0 era, the Internet is becoming more popular globally. Following that, the services it provides are gradually expanding, including the online news sector. Online newspapers help us update information around the world quickly, but their consequences are just as great. One of the hallmark of the age of the

Internet is the proliferation of fake news. They show up in high frequency across social platforms, blogs, unofficial news websites, etc. Fake news has made it difficult for us to determine the legitimacy of online news, oftentimes, causing critically false understanding of the mass on the nature of the world. For that reason, we need a tool to distinguish between real and fake news.

II. RELATED WORK

Kudari et al. [1] combined textual content and article subject to detect fake news from online social network and, using two methods of vectorization that are TD-IDF and CountVectorizer, with two classifiers that are Passive Aggressive Classifier and Naïve Bayes Classifier, obtained 90% accuracy. On the Spanish speaking side, Kun Li [2] used TF-IDF feature extraction technology and Stacking ensemble learning method based on five classifiers to classify Spanish COVID-19 news. This method achieved 75.48% accuracy and won second place in the FakeDeS Iberlef 2021 competition.

On the deep learning approach, Ilie et al. [3] employ three embedding methods for preserving word context that are Word2Vec, FastText and GloVe, with ten deep learning architectures to train a large dataset of 100000 article of ten labels (one Real label and the rest are different types of Fake). And reached an accuracy of 87,56% on the RCNN model with Specific Word2Vec trained on Lemma Text Preprocessing. And for Vietnamese, Heu et al. [4] used TF-IDF, PhoBERT and Character Counting embedding methods with five tree-based classification models, resulting in an accuracy of 95,21%.

III. DATA PREPARATION

This experiment is split into English and Vietnamese. For the English language, we use the Kaggle dataset and other datasets from large news websites. The Kaggle dataset from 2017 contains 45000 newspapers evenly divided between real and fake, the real news come from Reuters website whereas fake news are from websites deemed unreliable by Wikipedia. Other datasets come from different sources such as CNN, BBC, FOX, etc. But similar in date and distribution of real and fake. Together with the Kaggle dataset, this collection of data contains nearly 90000 newspapers.

For the Vietnamese test we use the Vietnamese Fake News Dataset (VFND) [5]. This contains over 200 news, collected in the period from 2017 to 2019 and uses several, cross-referencing sources, classified by the community.

IV. METHODOLOGY

This section is comprised of four parts: data preprocessing, word embedding methods, word segmentation method and data classification model.

A. Data preprocessing

Data preprocessing is an important first step for any machine learning tasks. It helps eliminate unnecessary features and filter noises, which can greatly impact performance. For our datasets, first, we merge title with content, convert all labels to 1 (Real news) or 0 (Fake news) and remove other unwanted columns. This allows us to merge different sets together as one big corpus. Then for each text in this corpus, we remove stopwords, i.e., commonly known words which add no meaning. We also use regular expressions to remove links, special characters, punctuations, etc. Finally, we eliminate any duplications that may occur from combining different datasets of different origins.

B. Word embedding

Word embedding is a natural language processing technique where each word or phrase is represented in a form of a real-valued vector that encapsulated its meaning. For this experiment we take two different approaches of statistic-based and context-based. For statistic-based approach, we use two techniques: TF-IDF Vectorizer and CountVectorizer. And for context-based approach, we use Word2Vec [6].

1. CountVectorizer

One of the simplest ways of doing vectorization. This technique builds a vocabulary of every unique word and generates a matrix of each word’s occurrence statistics across all documents.

2. TF-IDF

TF-IDF is a method used to determine the statistical significance of each word in the dataset. It is made up of two components: TF and IDF.

a. TF (Term Frequency):

The frequency of word in the document. For a specific word, it denotes the ratio of the number of times such word appears in the document to the word length of such document.

$$Tf(t,d) = \frac{number\ of\ t\ ocurrence\ in\ document\ d}{total\ word\ count\ of\ dcument\ d} \quad (1)$$

b. IDF (Inverse Document Frequency):

Measure the important of each word in the entire corpus, denoted by how common a word is across all documents.

$$Idf(t) = \log_e \frac{total\ number\ of\ documents}{number\ of\ documents\ that\ contain\ t} \quad (2)$$

c. TF-IDF (Term Frequency- Inverse Document Frequency):

Assign low weight to common repeating words, but high weight to word with high repetition in certain document. Which is equal to the multiplication of TF (1) and IDF (2).

$$TfIdf(t,d) = Tf(t,d) * Idf(t) \quad (3)$$

3. Word2Vec

Word2Vec is an unsupervised machine learning method that takes in the corpus without any label needed and output each word as a vector space, positioned such that word of similar meaning are closer to each other. Which allows for better understanding of the context from its meaning. Internally, Word2Vec leverages the use of two classification models to get the embeddings: Continuous bag-of-words (CBOW) and Skip-gram.

a. CBOW model:

This is a deep learning model takes the context of the input and try to predict the missing target word, corresponding to the context. This allows for better prediction of the target than Skip-gram (Figure 2).

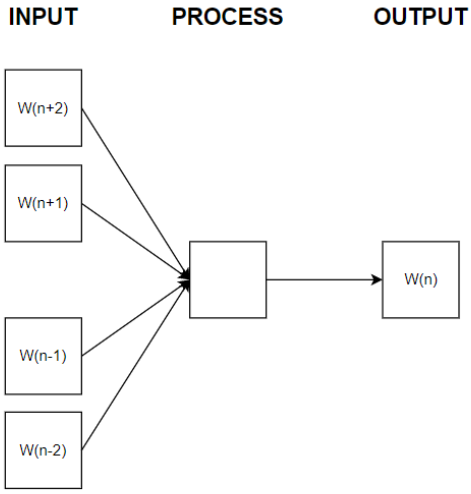

Figure 1. The CBOW model

b. Skip-gram model:

This is used for defining the context of a word. Given a target word, this model will predict the context words surrounding the target, opposite to how CBOW functions. This allows for better result for rare words in the corpus than CBOW (Figure 1).

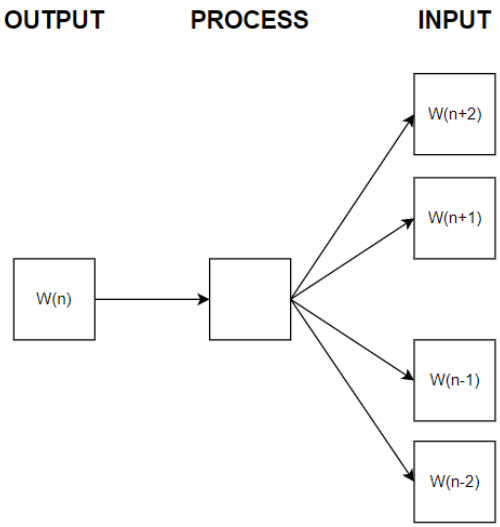

Figure 2. The Skip-gram model

C. Word Segmentation

Word segmentation is a process of splitting a sentence into many constituent parts, i.e., tokens. It is utmost important for every natural language processing task. For languages such as English, segmentation can be as simple as splitting at boundaries like white spaces and punctuations. However, for languages such as Vietnamese, boundaries are based on morphology, syntax, semantics, etc. Thus, requiring more complex segmentation approaches. To accomplish such task, we use Conditional Random Field (CRF) [7,8] and IOB (inside-outside-beginning) labelling [9].

1. CRF model:

CRF is a classification model for sequences of data, it models the dependency between each state and the whole input sequences. In this paper, we refer to CRF as the linear chain of states, which is a type of CRF that outputs sequences, corresponding to the sequences of input.

Let $x = (x_1, x_2, x_3, \ldots, x_T)$ be our input sequence, in this case, words to be labelled, and $y = (y_1, y_2, y_3, \ldots, y_T)$ be our state sequence. Let Y be a set of states, each of which is associated with a label $l \in L$. Then the CRFs can be defined as the conditional probability of a state sequences given the input sequences.

$$p(y|x) = \frac{1}{Z(x)} \exp\left(\sum_{t=1}^{T} \sum_k \lambda_k f_k(y_{t-1}, y_t, x, t)\right) \qquad (4)$$

Where $Z(x) = \sum_{s'} \exp\left(\sum_{t=1}^{T} \sum_k \lambda_k f_k(y'_{t-1}, y'_t, x, t)\right)$ is normalization summing over label sequences. λ_k is the learned weight of feature f_k. Which, in turn, is a feature function of maximum entropy modeling, either a transitional or per-state feature.

$$f_k^{(transitional)}(y_{t-1}, y_t, t) = \delta(y_{t-1}, l)\delta(y_t, l) \qquad (5)$$

$$f_k^{(per\_state)}(y_t, x, t) = \delta(y_t, l) b_k(x, t) \qquad (6)$$

Where δ denotes the Kronecker-δ. A per-state feature (6) combines the label l of current state y_t and function $b_k(x, t)$ that captures a particular property of the sequence x at time position t.

2. IOB:

A tagging scheme for labelling tokens into chunks in computational linguistics such as segmentation. It comprises of three tags. The B-prefix tag denotes that the token is the beginning of a chunk, the I-prefix is when a token in within a chunk, and the O tag signifies that a token belongs to no chunk at all.

The output of CRF will be in IOB format which can then be segmented accurately.

D. Classification model

For this paper, we use a Passive Aggressive Classifier [10] to classify the corpus. This belongs to a class of machine learning models known as online learning where input data comes in sequential order. The name Passive-Aggressive is based on its working tendency during training. When the model makes a correct prediction, it keeps all parameters unchanged, i.e., Passive. On the other hand, when a wrong prediction was made, it adjusts the parameters to finetune prediction, i.e., Aggressive. Due to its online learning category as well as the inherent passive tendency, this model is suitable for large dataset where batch processing of entire corpus is computationally heavy.

V. RESULT

This test is carried out for three different word embedding techniques on the English dataset. Performance will be measured based on percentage of correct predictions in all predictions.

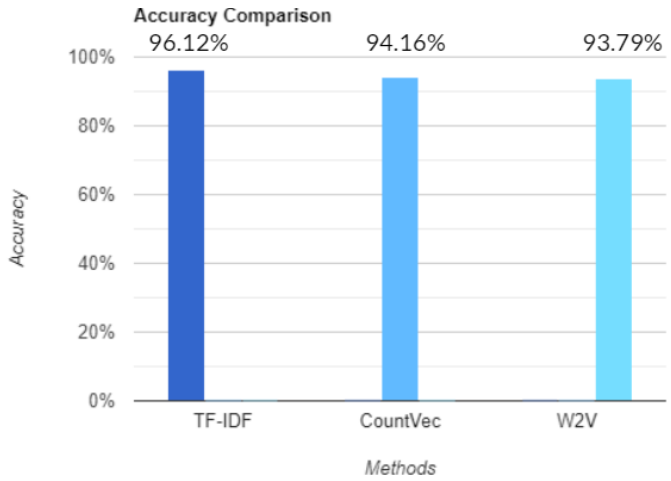

Figure 3. Accuracy comparison of embedding techniques

From Figure 3, we can see that the TF-IDF method produces the best result of 96.12%. Followed up by the CountVectorizer method at 94.16%. And lastly, Word2Vec at 93.79%.

The poor performance of Word2Vec can be explained by the limitation on how it is used in this classifier. Word2Vec embeds each word into a vector space, turning documents into sequences of vectors. However, the Passive Aggressive Classifier can only take in sequences of values. So, to make Word2Vec compatible with Passive Aggressive Classifier, me must take a vector of 1 feature and convert to a value, negating the benefit of Word2Vec over other methods.

The lower performance of CountVectorizer compared to TF-IDF is because of its simplicity. It only uses Term Frequency, treating all words equally, unlike TF-IDF which utilizes Inverse Document Frequency.

For our Vietnamese dataset, we performed two different experiments using TF-IDF Vectorizer. One with CRF segmentation, the other with simple white space segmentation.

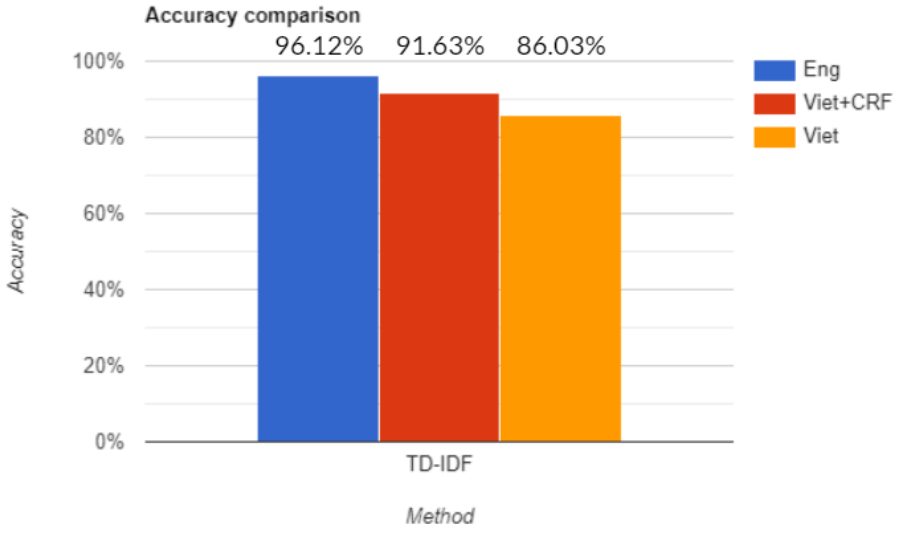

Figure 4. Accuracy comparison of CRF segmentation on Vietnamese dataset

Due to the small size of the Vietnamese dataset, we are met with an underfitting problem, causing performance to vary greatly between many tests and remained lower than the

better-fitted English test (Figure 4). But on average, the Vietnamese test with CRF segmentation performs better than one without at 91.63% and 86.03% respectively.

VI. CONCLUSION

Development of fake news is attracting people's attention because the problem of fake news is still spreading rapidly over the Internet environment. What is important, in this regard, is to classify the article whether as fake or real. In this paper, we have studied and built a machine learning model that can predict and classify articles. With the use of TF-IDF algorithm together with Passive Aggressive Classifier, the classification results have higher accuracy than other algorithms. For future work, we may build a website to make it easier for people to use this disinformation detection model, along with improving the accuracy of classification of articles written in Vietnamese.

References

[1] Kudari, J.M., Varsha, V., Monica, B., & Archana, R.J. (2020). Fake News Detection using Passive Aggressive and TF-IDF Vectorizer.

[2] Kun Li (2021). HAHA at FakeDeS 2021: A Fake News Detection Method Based on TF-IDF and Ensemble Machine Learning. In Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021) co-located with the Conference of the Spanish Society for Natural Language Processing (SEPLN 2021), XXXVII International Conference of the Spanish Society for Natural Language Processing., Málaga, Spain, September, 2021 (pp. 630-638). CEUR-WS.org.

[3] Ilie, V.I., Truică, C.O., Apostol, E.S., & Paschke, A. (2021). Context-Aware Misinformation Detection: A Benchmark of Deep Learning Architectures Using Word Embeddings. *IEEE Access, 9, 162122-162146.*

[4] Hieu, T., Minh, H., Van, H., & Quoc, B. (2020). ReINTEL Challenge 2020: Vietnamese Fake News Detection using Ensemble Model with PhoBERT embeddings. In *Proceedings of the 7th International Workshop on Vietnamese Language and Speech Processing* (pp. 1–5). Association for Computational Lingustics.

[5] Ho Quang Thanh, ninh-pm-se. (2019). thanhhocse96/vfnd-vietnamese-fake-news-datasets: Tập hợp các bài báo tiếng Việt và các bài post Facebook phân loại 2 nhãn Thật & Giả (228 bài).

[6] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of Workshop at ICLR, 2013.*

[7] Lafferty, John & Mccallum, Andrew & Pereira, Fernando. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. Proceedings of the Eighteenth International Conference on Machine Learning. 282-289.

[8] Cam-Tu Nguyen, Trung-Kien Nguyen, Xuan-Hieu Phan, Le-Minh Nguyen, and Quang-Thuy Ha. 2006. Vietnamese Word Segmentation with CRFs and SVMs: An Investigation. In *Proceedings of the 20th Pacific Asia Conference on Language, Information and Computation*, pages 215–222, Huazhong Normal University, Wuhan, China. Tsinghua University Press.

[9] Ramshaw, L., & Marcus, M. (1995). Text Chunking using Transformation-Based Learning. In *Third Workshop on Very Large Corpora*.

[10] Crammer, Koby & Dekel, Ofer & Keshet, Joseph & Shalev-Shwartz, Shai & Singer, Yoram. (2006). Online Passive-Aggressive Algorithms. Journal of Machine Learning Research. 7. 551-585.

Data & Source code

Item	Link
Dataset	Fake and Real news dataset (Kaggle) Fake News finder (Github) Vietnamese Fake news dataset (Github)
Source code	Fake news detector (Github)

Self Assessment

Criteria	Thai Duy Bao	Huynh Minh Triet
Punctuality	10	10
Hard Working	10	10
Cooperation	10	10
Contribution	10	10
Summary	**10**	**10**