CS4248 Final Report

A0219866M, A0219816X, A0201965E, A0200053E, A0218330R, A0219888E
Group 08
Mentored by Hai Ye
{e0550524, e0407034, e0544366, e0550474, e0415774, e0550546}@u.nus.edu

Abstract

We present a project to categorize news articles gathered and labelled by H. Rashkin et al. (2017) into "Satire", "Hoax", "Propaganda", and "Reliable News" sub-types to help combat the proliferation of unreliable news. We compare many different models to see which is most effective at performing this task. Our experiments show that using AI to discern truth from fiction may be more feasible than one might expect, though more can still be done for further classification into sub-categories.

1 Introduction

As a global society able to connect with every corner of the world through a simple click of a button, we are forced to interact with and process an unprecedented amount of information daily. With this in mind, it is no surprise that most consumers choose not to conduct extensive research on every article they come across, instead preferring to take the information at face value. However, that may not always be the best option.

According to a survey done by Ipsos[1], only 1 in 10 Singaporeans are able to avoid being fooled by fake news headlines, even though 80% of those asked were confident in their abilities to tell real from fake. Our team finds this statistic extremely worrying as misinformation like this can cause hysteria, distrust, or even incite hate[2].

Dedicated organisations such as Poynter and PolitiFact strive to increase media literacy amongst readers by actively investigating the legitimacy of news articles and breaking down each step of their process while doing so. This is incredibly admirable work. However, this is an extremely time-consuming process, making it an impossible task to keep up with the roughly 3 million news articles published every single day [3].

In this paper, we show our attempts to address this issue using Natural Language Processing models which categorize articles solely based on their content. We also show how the results of this model can be interpreted by a human user to get a better understanding of why our models are making the choices they do.

For all models created during our experimentation, our inputs are some vectorised version of the news article (specifics to be discussed in later sections) and our outputs are a single label ranging from 1 to 4 (representing "Satire", "Hoax", "Propaganda", and "Reliable News" respectively).

This report will include the following sections:

1. Introduction
2. Related Work/Background
3. Corpus Analysis & Method
4. Experiments
5. Discussion
6. Conclusion

2 Related Work / Background

Many publications (Thorne et al., 2018; Zhou et al., 2019; Zhong et al., 2020) concentrate on the related issue of fact checking, which tries to find evidence from outside knowledge to verify the veracity of a claim (e.g., a subject-predicate-object triple). In general, fact-checking is more comprehensive while fake news detection typically concentrates on news events (Oshikawa et al., 2020). Fake news detection consists of two categories: social-based fake news detection and content-based fake news detection.

[1] https://www.ipsos.com/en-sg/susceptibility%2Dsingaporeans%2Dtowards%2Dfake%2Dnews

[2] https://www.who.int/europe/news/item/01-09-2022%2Dinfodemics%2Dand%2Dmisinformation%2Dnegatively%2Daffect%2Dpeople%2Ds%2Dhealth%2Dbehaviours--new%2Dwho%2Dreview%2Dfinds

[3] https://earthweb.com/how%2Dmany%2Dnews%2Darticles%2Dare%2Dpublished%2Devery%2Dday/

Social-based fake news detection refers to involving news reports contains comprehensive data to help identify bogus news, such as user profiles and social interactions. An example of social-based fake news detection is a heterogeneous network of news articles, creators, and subjects, which consequently presented a deep diffusive network model for incorporating the network structure information (Zhang et al., 2020).

Conversely, content-based fake news detection refers to using clues contained in the news content to differentiate fake and trusted news. Many studies extract particular writing styles, such as lexical and syntactic elements (Shu et al., 2020; Oshikawa et al., 2020).

Related Solutions

Based on others' experiences in detecting fake news, Support Vector Machine, and Naive Bayes are the more commonly used models for simple fake news detection (Shu et al., 2020), where the models' only objective is to indicate if a headline is fake or real. These two models differ a lot in structure, though both usually act as baseline models for further development. Logistic regression (LR) (Bhattacharjee et al., 2017) and variations of Decision Tree such as Random Forest Classifier (RFC) (Hassan et al., 2017) are also used occasionally.

CompareNet (Wu et al., 2019) is one example which directly compares the news to external knowledge for fakes news detection, and it achieved a F1 Score of 0.6826 for a 4-way classification.

Tokenization, stemming, and generalization or weighting of words are frequently used in preprocessing. Term Frequency-Inverse Document Frequency (TF-IDF) is widely employed to transform tokenized texts into features. Pre-learned word embedding vectors such as word2vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014) are also frequently utilized to vectorize word sequences.

3 Corpus Analysis & Method

3.1 Preprocessing

Exploration of preprocessing was done in reference to what was done by predecessors (Chen et al. 2017). Some of the preprocessing includes stop word removal, contraction splitting, lemmatization, punctuation removal, lowercasing texts, and tokenization. Most preprocessing steps are done in order to transform the text into a clean and consistent format. For instance, removing stop words helps build a cleaner dataset as well as reduce the dataset size. Furthermore, lemmatization is used in order to treat words that should be the same similarly, i.e. national and nation. Punctuation is also removed to help treat each text equally. Lastly, tokenization aims to split sentences into smaller units such that they can be more easily assigned meaning.

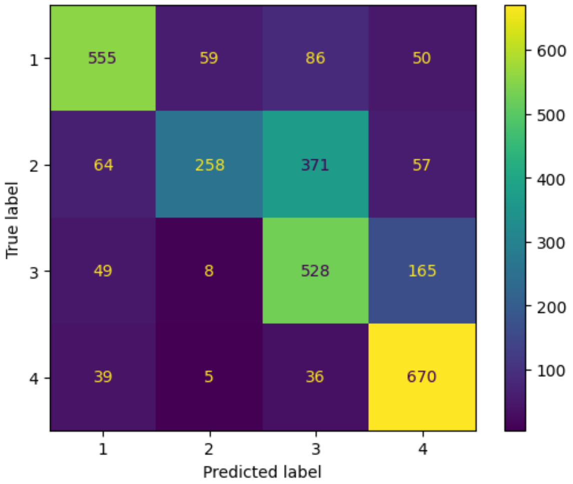

Figure 1: Confusion Matrix of Logistic Regression

3.2 Baseline Model

As previously mentioned, commonly used models for baseline models include Support Vector Machine, Naive Bayes, or Logistic Regression. After experimenting with these models while keeping our preprocessing steps consistent, it was found that Logistic Regression performed best with a test accuracy of 67% as seen in Figure 1. Specifically for this Logistic Regression model, we used stop word removal, lemmatization, and count vectorization to derive the test accuracy mentioned before.

3.3 Data Collection

The dataset utilized was originally constructed by Rashkin et al. (2017), and was used in an analytical study on the language of news media in the context of political fact-checking and unreliable news detection. The data was collected from various sources such as the Onion, American News, The Activist, and the Gigaword News. The dataset was then split into four categories: "Satire", "Hoax", "Propaganda", and "Reliable News".

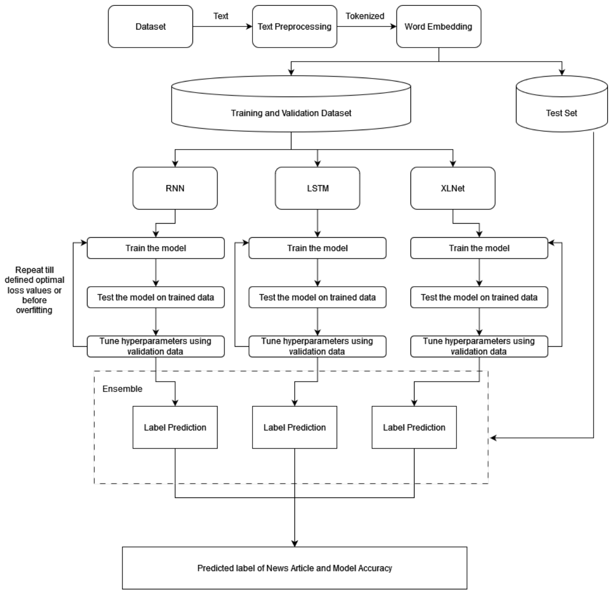
Dataset
Text
Text Preprocessing
Tokenized
Word Embedding
Training and Validation Dataset
Test Set
RNN
LSTM
XLNet
Train the model
Test the model on trained data
Tune hyperparameters using validation data
Repeat till defined optimal loss values or before overfitting
Ensemble
Label Prediction
Predicted label of News Article and Model Accuracy

Figure 2: Proposed unreliable news detection model

3.4 Data Analysis

The dataset constructed (Rashkin et al., 2017) was analyzed by the team. It was observed that the dataset was skewed as seen in Table 1, i.e. some categories consist of a significantly larger number of dataset. Hence, F1 score is used in order to find the weighted average of precision and recall. This provides a better insight when evaluating skewed dataset as compared to relying solely on accuracy scores.

3.5 Main Methods

We conducted feature engineering to extract meaningful information from raw data to make it usable for our machine learning models. We also explored various deep-learning methods including Recurrent Neural Networks (RNN) with embeddings, LSTM with Attention, and XLNet. Figure 2 describes our proposed unreliable news detection model, which contains an ensemble of the three previously-mentioned models. For this final model, we managed to achieve an accuracy of 77%.

Feature Engineering

Before extracting features, the basic steps of preprocessing were applied. These include stop word removal, lowercasing texts, lemmatization, and contraction expansion. After further analysis, the features we decided to extract were word count, average word length, average sentence length, punctuation count, number of second-person-pronouns, and sentiment analysis. Our plan was to use these extracted features during the training of our models. However, we noticed that when trying to incorporate these features, our test accuracy either dropped or had no significant change. This resulted in us shifting focus to finding a better performing model for the dataset.

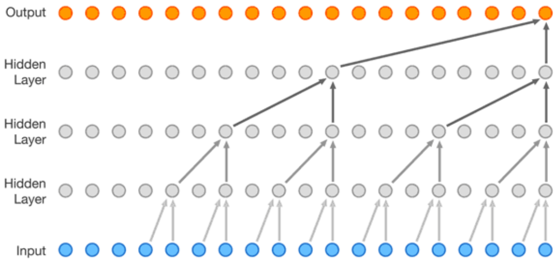

Figure 3: Feed forward model XLNet

RNN

RNN with multiple return sequences enables the model to learn more complex sentences, mimicking that of LSTM. However, it is more susceptible to both exploding and vanishing gradient problems. RNN is thus explored as we would like to compare the difference in the lack of forget gates and its effects, since LSTM is usually considered to yield better results.

LSTM

(Bidirectional) LSTM is chosen to capture information from both directions (past and future data), allowing for a more robust understanding of the context surrounding each word in the text. Adding attention later to the Bi-LSTM model allows it to focus on the most relevant parts of the input text, which can help reduce the noise and improve the model's performance.

XLNet

XLNet attempts to make an improvement on the BERT model by enabling learning bidirectional context. XLNet also uses a autoregressive model which uses the k most recent inputs to predict y (Yang et al., 2020). This is also known as a feed-forward model, as shown in Figure 3. Upon testing of the base-cased XLNet model against base-cased DistilBERT, which performs knowledge distillation to make the BERT model faster and lighter, it was discovered that XLNet outperformed DistilBERT, as shown in Table 2. Therefore, XLNet is one of the three models chosen for ensembling.

4 Experiments

4.1 Experimental Settings

News Type	Source	No. of Docs
Satire	The Onion The Borowitz Report Clickhole	14797
Hoax	American News DC Gazette	7692
Propaganda	The Natural News Activist Report	18620
Reliable News	Gigaword News	10745

Table 1: Statistics of dataset used in training, validation and evaluation.

The LUN: Label Unreliable News dataset (Rashkin et al., 2017) was used in our experiments for building the Logistic Regression baseline model and various deep-learning models. Details of the dataset can be found in Table 1.

We split the obtained data into train and test sets, setting aside 3000 news articles for testing with each label having 750 documents. Furthermore, we used 20% of the training data, approximately 10000 articles, as a validation set while developing our deep-learning models.

Our goal is to classify these news articles from into one of the four aforementioned categories with F1 score as our primary evaluation metric. This metric was chosen due to the multiple classes in our dataset, and the F1 score allows us to combine the precision and recall scores of our models.

4.2 Deep-learning Models

We initially tested multiple models to determine which performed the best, and subsequently which we should focus our attention on. These included Support Vector Machines (SVM) with Continuous Bag Of Words (CBOW), XLNet, RNN with word embeddings, Term Frequency - Inverse Document Frequency (TF-IDF), DistilBERT, LSTM with attention, and Temporal Convolutional Network (TCN).

All of these models were trained on preprocessed text as was discussed in Section 3.1. Tuning hyperparameters such as increasing batch size, the use of dropout layers, regularization penalty, and a lower optimizer learning rate did little to improve the models' accuracy. A summary of the performance of all models is shown in Table 2, where TCN, TF-IDF, CBOW with SVM, DistilBERT, BERT Embedding with Logistic Regression, text with LSTM and Word2Vec Embedding with LSTM obtained results of 40%, 50%, 51%, 54%, 60%, 61% and 63% respectively. These models did not manage to surpass the baseline's model accuracy and as such, more emphasis was placed on LSTM, RNN and XLNet in order to further improve and outperform the baseline model.

Model	Features	Accuracy
TCN	Text	.40
TF-IDF	Text	.50
CBOW + SVM	Text	.51
DistilBERT	Text	.54
Logistic Regression	BERT Embedding	.60
LSTM	Text	.61
LSTM	Word2Vector Embedding	.63
Baseline Model	**Text**	**.67**
LSTM	Text + Attention	.67
RNN	Text + Embeddings	.69
XLNet	Text	.71
Ensemble	Text	.77

Table 2: Model performance on test set.

4.3 Ensembling

Method	Accuracy
Voting	.75
Aggregation	.77

Table 3: Ensemble methods and their accuracies.

Model	Label 1	Label 2	Label 3
1	0.4	0.5	0.1
2	0.4	0.5	0.1
3	0.4	0.1	0.5
Aggregate	0.4	0.37	0.23

Table 4: Example of outlier handling for aggregation.

As tweaking the different models did not yield significant improvements, we decided to venture into ensembling the models. There are two main methods to ensemble: voting and aggregation (Singh, 2018). Ensembling was done on the three best performing models: LSTM, RNN, and XLNet. For voting, each model votes on the label with the highest probability to a specific text and the label with majority votes is chosen. Tiebreakers are decided randomly. For aggregation, the probabilities for each label produced by the models are aggregated together before making the decision on the correct label for the text. The final probability matrix would then be the average of all three best performing models. The accuracies for these two methods are shown in Table 3.

It can be seen that both methods produced similar results, with aggregation edging out voting by a small margin. We decided to use the aggregation method because it manages outliers better than the voting method. One example is shown in Table 4. It can be observed that the highest aggregate probability is Label 1, even though none of the models originally predicted Label 1 to have the highest probability. If the voting method were to be used, Label 2 would be predicted as models 1 and 2 voted for it. From this example, we can observe that aggregation focuses on the results from all the models while voting merely focuses on the highest probabilities, which can be detrimental if there were to be outliers in the predictions. The viability of the voting method improves when the number of models used for ensemble increases.

4.4 Final Results

After our experiments, we ultimately chose to employ an ensembling approach for the three top performing models, as we concluded that making adjustments to individual models did not yield significant improvements, and was still sub-optimal. This final ensemble model achieved an accuracy of 77%, which is a considerable improvement from the baseline model. Thus, our findings suggest the effectiveness of ensembling in improving the overall performance of the models.

	Precision	Recall	F1-score
Satire	.90	.83	.86
Hoax	.90	.59	.71
Propaganda	.65	.69	.67
Reliable News	.72	.97	.82
Accuracy			.77
Macro avg	.79	.77	.77
Weighted avg	.79	.77	.77

Table 5: Ensemble model performance of 4-way classification.

5 Discussion

5.1 Research Questions

This study focuses on two research questions that aim to explore the applicability of Natural Language Processing (NLP) techniques in detecting unreliable news, as well as to evaluate the effectiveness of ensemble models used in this study.

The first research question is centered on assessing the applicability of NLP techniques to detect unreliable news, which has shown promising results on the test dataset. However, the ability of these models to detect unreliable news in real-world scenarios must be considered, as the models draw inferences from linguistic features rather than fact-checking. As fake content continues to evolve, our current training data might no longer contain the same complexity and nuance present in the newer news articles, rendering our models outdated.

Our second research question pertains to the NLP aspects of the models used in this study. We explored how well the models perform in identifying the linguistic features of the text that contribute to the classification of news articles. In the following sections, we will conduct further study to understand which text features are most relevant in identifying different types of unreliable news, including Satire, Hoax, Propaganda, and Reliable news articles. This analysis is essential for understanding the underlying mechanisms of our models.

5.2 Macro Analysis

Upon viewing the confusion matrix of multiple model predictions, we identify Propaganda to be the worst-performing class among all 4 classes based on the F1-score. The misclassification of most labels is also related to this class.

Firstly, a significant portion of Hoax text was predicted to be Propaganda text in several models. This is in line with our observations during initial data exploration, where we often found it difficult to differentiate between the Hoax and Propaganda data. The intentions of writing the news segments from these two classes largely differ, yet their common purpose is to push a narrative that is able to convince most people. To label a text as a Hoax or Propaganda would depend on personal biases and perspectives. Thus, we propose to combine these two classes as an Unreliable class in further studies to reduce ambiguity when labelling the training set.

Next, a significant portion of Propaganda text was predicted to be Reliable news in several models. This is unsurprising because our project is based on the assumption that we are able to filter unreliable news by identifying unique semantic features. However, we need to incorporate factual evidence to differentiate propaganda and reliable news in some cases. For example, two articles on the same issue could be categorized into different classes when we consider personal biases.

This two types of misclassification are reflected in the precision and recall. The misclassification of Hoax as Propaganda results in lower precision in the Propaganda class and lower recall in the Hoax class, while the missclassification of Propaganda as Reliable News results in lower precision in the Reliable News class and lower recall in the Propaganda class.

Disregarding these metrics, the model shows promising results for the other unaffected labels, with precision and recall scores ranging from 0.83 to 0.97.

5.3 Micro Analysis

In addition to analysing the model performance at the class level, we consider the interpretability of our models to answer research question 2. To better understand how our models make decisions and provide insights into tackling fake news, we employed the Shapley Additive Explanation(SHAP)[4] technique.

The SHAP explanation method is based on coalitional game theory, which enables the computation of Shapley values that disclose the individual contribution of each player (or feature) on the model's output. This information is particularly important, as it can help to bolster user confidence in the model's predictions and inform decision-making on how to best mitigate the impact of unreliable news.

Local Explanation Example

Given this text:

"*The United States of America, a nation with a population of approximately three hundred million people, totally accepts that the next President of the United States can only be selected from two families. In interviews conducted across the country, Americans acknowledged that, while the United States boasts many exceptional people in the fields of technology, business, public policy, and government, none will be offered to voters as candidates because they do not come from one of the two families deemed eligible. No doubt about it, there are a lot of great people out there who could be President, said Stoddard Vinton, of Toledo. But I guess our system of choosing people from just two families has worked out pretty well. Leslie McEdwards, of San Jose agreed, that, while it would be cool to choose a President from more than two families, on the plus side, we voters dont have to learn a bunch of new names. This country is facing unprecedented problems, and its going to take some fresh ideas to solve them, said Doug Chessing, of Grand Rapids. Ive got my fingers crossed that someone from one of those two families can do it. The fact that the current President, Barack Obama, belonged to neither of the families always felt kind of weird to me, said Halynn Cross, of Knoxville. He tried really hard and all, but, after eight years, itll be nice to get back to someone from the two families. In one of the strongest endorsements of Americas two-family system, Rick Keelins of Albany said that he is sick and tired of people complaining about it. At least we have two families to choose from, he said. A lot of countries, like North Korea, just have one.*",

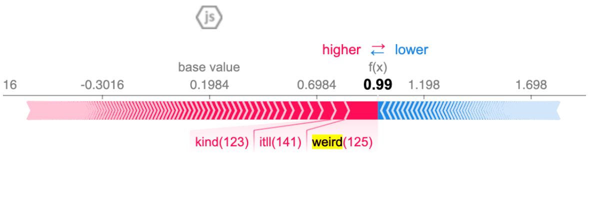

Figure 4: LSTM Local Explanation with SHAP

LSTM

The LSTM model will predict the text as satire news, which is the correct label. As shown in Figure 4, the local explanation for this prediction is largely caused by the presence of 'weird' in 'The fact that the current President, Barack Obama, belonged to neither of the families always felt kind of weird to me, said Halynn Cross, of Knoxville.'.

Although it is not very obvious, we can agree that this quote contributes to the text's satirical tone by highlighting the absurdity of the argument that presidential candidates can only come from two families. The word "weird" suggests that the idea that a president could come from outside these two families is unusual or unexpected, which is obviously untrue. This use of exaggeration is a common technique in satire, and it helps to emphasize

[4] https://github.com/slundberg/shap

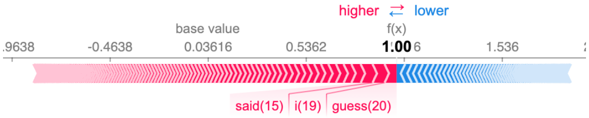

Figure 5: RNN Local Explanation with SHAP

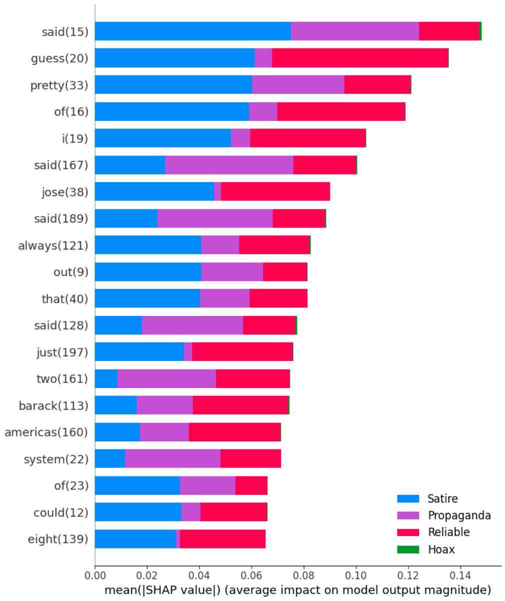

Figure 6: RNN Local Explanation Summary

the absurdity of the argument being made in the text. Overall, the inclusion of ‘weird‘ reinforces the satirical nature of the text.

RNN

As for the RNN model, even though it labels the news as satire too, their local explanations are quite different, as shown in Figure 5 and 6. In the RNN model, ‘guess‘ is the word that provides the strongest signal for ‘satire‘ class. The sentence "But I guess our system of choosing people from just two families has worked out pretty well." contributes to the satirical tone by presenting an absurd and untrue premise as if it were a reasonable statement. The use of the sentence is pretty ironic, as it implies that the current political system is successful and effective when in reality it is being criticized for its limitations and flaws.

5.4 Interpretability

While the SHAP explanation method provides a valuable means of interpreting the output of our models, the interpretability offered by this technique is still limited. The key features or words identified by SHAP might not always be straightforward, as SHAP does not automatically capture the relationships between words and the context in which they appear. This is particularly true in the case of deep neural networks, where each word is often related to its neighboring words and sentences in complex ways. Therefore, additional human inferences are usually necessary, though this could potentially introduce biases into the interpretation.

5.5 Model Ensemble

Based on the discussion above, it is evident that each model has a unique interpretation of the relationship between texts and their corresponding classes. Therefore, we decided to leverage the strengths of these three different models - XLNet, RNN and LSTM in our ensemble. While each of these models had achieved only modest improvements over the baseline model, their combination compensates for each other’s knowledge gap, resulting in a higher accuracy of 75-77%.

Another benefit of the ensemble model is it can mitigate the risk of overfitting as it removes the total reliance on a single model. By combining several models that were trained on different architectures, we created a more robust model capable of generalizing better to new data.

Our ensemble model showcases remarkable performance when compared to the CompareNet implementation (Hu et al., 2021), a state-of-the-art model that has previously demonstrated superior performance in this task. Our ensemble model has outperformed CompareNet in all metrics, showing promising results in detecting unreliable news.

6 Conclusion

We have created a human-interpretable model able to categorize news articles based on their "intent" and caution readers to do further research into a topic.

Although we are proud of the progress of our project[5], one major limitation of our model is that our model labels texts based on inferences from linguistic features, rather than proper fact-checking. Therefore, this system runs on the assumption that there exists a linguistic difference between statements which are false and statements which are true.

Additionally, the current iteration of our model has mostly been trained on political quotes as well

[5] https://github.com/ngchisern/cs4248%2Dproject

as news media. As such, it is most accurate when classifying articles of a similar nature. Future variations of this model could instead be trained using a dataset consisting of social media posts. Such a model would be able to assist users on social media in avoiding fake news by flagging posts it deems unreliable and prompting users to do more research.

References

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Sreyasee Das Bhattacharjee, Bala Venkatram Balantrapu, and Ashit Talukder. 2017. Active learning based news veracity detection with feature weighting and deep-shallow fusion.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions.

Linmei Hu, Tianchi Yang, Luhao Zhang, Wanjun Zhong, Duyu Tang, Chuan Shi, Nan Duan, and Ming Zhou. 2021. Compare to the knowledge: Graph neural fake news detection with external knowledge. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 754–763, Online. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Hassan N., Arslan F., Li C., , and Tremayne M. 2017. Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster.

Ray Oshikawa, Jing Qian, and William Yang Wang. 2020. A survey on natural language processing for fake news detection.

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, Doha, Qatar. Association for Computational Linguistics.

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937, Copenhagen, Denmark. Association for Computational Linguistics.

Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. Fakenewsnet: A data repository with news content, social context and spatialtemporal information for studying fake news on social media.

Aishwarya Singh. 2018. A Comprehensive Guide to Ensemble Learning (with Python codes).

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Lianwei Wu, Yuan Rao, Haolin Jin, Ambreen Nazir, and Ling Sun. 2019. Different absorption from the same sharing: Sifted multi-task learning for fake news detection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4644–4653, Hong Kong, China. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2020. Xlnet: Generalized autoregressive pretraining for language understanding.

Jiawei Zhang, Bowen Dong, and Philip S. Yu. 2019. Fakedetector: Effective fake news detection with deep diffusive neural network.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Reasoning over semantic-level graph for fact checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6170–6180, Online. Association for Computational Linguistics.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. GEAR: Graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 892–901, Florence, Italy. Association for Computational Linguistics.

Acknowledgements

This document has been adapted from the ACL Rolling Review Template (ACL ARR) by Min-Yen Kan.

Statement of Independent Work

1A. Declaration of Original Work. By entering our Student IDs below, we certify that we completed our assignment independently of all others (except where sanctioned during in-class sessions), obeying the class policy outlined in the introductory lecture. In particular, we are allowed to discuss the problems and solutions in this assignment, but have waited at least 30 minutes by doing other activities unrelated to class before attempting to complete or modify our answers as per the class policy.

We have documented our use of AI tools (if applicable) in a following table, as suggested in the NUS AI Tools policy[6]. This particular document did not use any AI Tools to proofcheck and was constructed and edited purely by manual work.

Signed, [A0219866M, A0219816X, A0201965E, A0200053E, A0218330R, A0219888E]

[6] https://libguides.nus.edu.sg/new2nus/acadintegrity, tab “AI Tools: Guidelines on Use in Academic Work”