JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 37, 1177-1186 (2021)
DOI: 10.6688/JISE.202109_37(5).0013

Short Paper

Hyperpartisan News Classification with ELMo and Bias Feature

GERALD KI WEI HUANG AND JUN CHOI LEE
Faculty of Computer Science and Information Technology
Universiti Malaysia Sarawak
Kota Samarahan, 94300 Malaysia
E-mail: 19020037@siswa.unimas.my; jclee@unimas.my

Hyperpartisan news is a kind of news riddled with twisted, untruthful, and often extremely one-sided. This kind of news can spread more successfully than the others. One of the obvious traits of hyperpartisan news content is that it can mimic regular news articles. Most are favour fake news detection algorithms, and there is less research conducted for hyperpartisan news. This research aims to perform classification on the hyperpartisan news using ELMo and bias features. ELMo was used to develop a classification model to perform classification on the BuzzFeed Webis News Corpus dataset. The model uses ELMo embedding with bias word score generated from bias lexicon to train a deep learning model using Tensorflow and Keras. We had compared the final result with two proposed baseline models that utilized ELMo from other research. The discussion section further investigated the contribution of ELMo and bias feature in the hyperpartisan task.

Keywords: natural language processing, classification, hyperpartisan, ELMo, bias detection

1. INTRODUCTION

In the past, fake and bias news problem inspired various studies in identifying and classifying this kind of news. There has been concern that fake and bias news are misleading the readers. The content on the news can be entirely false or misleading. For better understanding and defining the terms, we can loosely classify the contents. The contents that solely misleading and fabricated are known as fake news. Fake news mimics news and mainly aims to deceive the reader. Another form of content is hyperpartisan news, which is biased or misleading and usually covers actual events or incidents. Fake news content can be entirely false, while hyperpartisan news can be a mixture of true and false. However, according to [1], hyperpartisan news is a kind of news riddled with twisted, untruthful, and often extremely one-sided. This kind of news can spread more successfully than the others. Currently, there is less research conducted for hyperpartisan news. [2] claims that fake news detection is still in its infancy stage, and a near real-time reaction is crucial. Hyperpartisan contents can lead to polarization within a community because it is typically used to manipulate propaganda and manipulate readers. When this situation goes to an extreme state, it can cloud readers judgements on making objective decisions. There are

Received March 16, 2020; revised July 3, 2020 & January 7 & May 4, 2021; accepted May 16, 2021.
Communicated by Lee Hung Liew.

many hyperpartisan news and articles to propagate biased news for political or financial gain.

Hyperpartisan news is written in a non-neutral manner and deviates from the truth by using dedicated words, bias words, or authors own preferred points of view. This research aims to identify and classify hyperpartisan news using ELMo (Embeddings from Language Models) [3] and bias features on the BuzzFeed Webis News Corpus dataset [2]. ELMo was selected because there are recently emerge and gain impactful ground in Natural Language Processing (NLP) community. ELMo has the ability to model complex characteristics of words used and how a word uses in various linguistic contexts (Polysemy). Instead of using pre-trained Global Vectors for Word Representation (GloVe) [4] or other word vectors, this research used ELMo produced word vectors. The ELMo vectors then used to train a classification model to identify and classify hyperpartisan news. Since hyperpartisan news is usually non-neutral and tends to be one-sided, an effort was added to detect the onesided terms and the bias-inducing lemmas in the news articles. Then we use the bias occurrences as a feature for the training. The dataset went through basic cleaning and minimal pre-pre-processing steps. The data cleaning process includes the removal of Hypertext Markup Language (HTML) tags and advertisements, text splitting using NLP toolkit, spaCy and scikit-learn. The ELMo use in this research is the pre-trained ELMo from Tensorflow Hub and use a classification model to perform the classification task. Finally, we compared the classification result with the results of two baseline models that similarly utilize ELMo from other research to evaluate the performance of ELMo and the bias features in the task mentioned above. The following section consists of a related work review, the introduction of the research methodology, the results and a discussion of the experiments mentioned earlier.

2. RELATED WORK

There are many previous studies and attempts in the detection and classification of misleading news and articles. The methods and approaches include cross-checking with known facts, deception detection, writer profiles and writing styles. One of the common approaches used for the detection is by relating to known facts [5], analyzing news spread on social media or veracity [6, 7] and even by analyzing the writing style of the writers or authors known as style-based [2, 8].

[9] applied the statistical web-based checking of textual documents approaches that measure the frequency of documents in supporting a claim. The authors used anchor point methods that comprise fact selection to extract the noun-to-noun facts from a document using the Part of Speech (POS) tagger. The extracted facts then went through individual fact assessment based on search engine results. The queries are weighted based on the occurrence frequency and Uniform Resource Locator (URL) ranking, followed by weighing the facts and generalized to aggregate a score for the document. Another effort has been made by [10] to apply the shortest path between concepts in a knowledge graph. The authors performed an evaluation by examining tens of thousands of claims related to history, entertainment, geography, and biographical information using a public knowledge graph extracted from Wikipedia. The authors framed the problem as a network problem. The shortest path between concept nodes and semantic proximity metrics on the knowledge graph technique was evaluated by the authors using tens of thousands of claims. The

results indicated that important and complex human fact-checking tasks could be effectively replaced and reduced to a network analysis problem.

[6, 7] applied Deception Detection approaches at scale to detect uncertainty in social media. [6] claimed that social media authors enjoy the free form of writing. To study the differences, the researchers annotated a small set of randomly sampled tweets according to the scheme of uncertainty types, then manually identified all the possible uncertain tweets. The results from [6] shown that content-based features have the highest improvement and identify uncertain tweets and uncertain cue-phrase. [11] attempts to determine the veracity of a claim based on the conversation on Twitter. The researchers propose a shared task where participants analyze rumors in the form of claims made in user-generated content. Users respond to one another within conversations attempting to resolve the veracity of the rumor. The tweet in a conversation thread can be classified into supporting, denying, querying and commenting for the tweet classification tasks. Most of the researchers were using deep learning models. For instance, Long Short Term Memory (LSTM) for sequential classification or Convolutional Neural Network (CNN) for getting the representation of each tweet, then compute a probability using a Softmax classifier. The results have shown that elaborating feature engineering and proposing a method to address the class imbalance in the tweets data sources gained a higher impact. Similar methods and features were used for the veracity classification task. However, the best performing model used the percentage of reply tweets classified as either support, deny or query as added features.

[12] proposed a hybrid model that uses one LSTM for obtaining the representation of news articles, and the other LSTM uses speaker profiles to obtain the vector representations of speakers. Then Soft-max function was used to concatenate the two for classification. This research shows that author information such as speaker profiles can be a helpful feature for fake news detection. [13] also used style features for fake news detection. The stylistic features are based on natural language processing to understand the syntax, text style, and grammatical elements of the content and title of the article. The researchers used the two well-known hypothesis testing methods, the one-way Analysis of Variance (ANOVA) test and the Wilcoxon rank-sum test, to determine which features differ between the different news categories. It is then further strengthened by using Support Vector Machine (SVM) classification results to classify fake and authentic news articles by their titles. The results showed that fake news titles use significantly fewer stop words and nouns. Instead, it significantly contains more proper nouns and verb phrases.

[2] performed classification on the BuzzFeed Webis News Corpus to classify the hyperpartisanship vs mainstream articles. Among the selected publishers, there are six prolific hyperpartisan: three left-wing, three right-wing, and the other three were mainstream. The authors proposed a writing style model that uses style features, consisting of n-grams of characters, stop words and part of speech with the n in [1, 3]. The authors further employ ten readability scores and dictionary features. Each one indicated the frequency of words from a tailor-made dictionary in a given document using General Inquirer Dictionaries. The domain-specific features consist of external links, number of paragraphs, average length in a document and ratios of quoted words. The authors trained three random forest classifiers to perform the task. The results have shown that hyperpartisan news articles can be distinguished from more balanced news by writing style alone. The researchers concluded that the writing styles of two opposing orientations were very similar, which ap-

peared to be a typical writing style of both left and right extremism.

[14] introduced an ELMo Sentence Representation Convolutional (ESRC) Network. The authors pre-calculate sentence level embeddings as the average ELMo word embeddings for each sentence and represent the document as a sequence of such sentence embeddings. The sentence embeddings are treated as a sequence input feed into a convolutional network (CNN) classifier. The authors also applied the Batch Normalization (BN) in the CNN to learn the document representations. The proposed ESRC Network can achieve an accuracy of 82.2% and become the Winner in Semeval 2019 Hyperpartisan task. This model is used as the first baseline.

[15] conducted research on the performance ELMo and Bidirectional Encoder Representations from Transformers (BERT) [16] on the hyperpartisan classification task. The research proposed two separate models that utilized ELMo and BERT, respectively. In the research, the dataset went through minimal data pre-processing to test the performance of ELMo. Then the authors used Tensorflow and Tensorflow Hub pre-train ELMo to produce the ELMo vectors. The vectors then serve as the input to train a Logistic Regression classifier to perform classification on the hyperpartisan news. Although the initial experiment showing the model does not benefit from the training, the improved model can reach an accuracy of 60.8 %. For this research, the ELMo experiment is used as the second baseline. BERT is not included in this research.

Hyperpartisan news article tends to be non-neutral manner. Often include the authors own perspective and view. [17] concluded his research and study suggested that the bias in his bias reference works generally fall under two categories, framing bias and epistemological bias. In our case, we are focus on framing bias. According to [17], framing bias can be subcategorized into sub bias known as subjective intensifiers and one-sided terms. Subjective intensifiers are adjectives or adverbs that add a subjective strength to stress the meaning of a phrase or proposition. One-sided terms often only reflect only one of the sides of a contentious issue. Detecting and identifying the existence of one-sided terms, intensifier, and bias-inducing lemmas can greatly help drawing a line to separate the hyperpartisan and mainstream articles. However, the bias feature alone is not enough.

3. METHODOLOGY

This section described and explained the stages and flow of our proposed model to study how the bias words and ELMo embeddings perform in the hyperpartisan classification task. The discussions also included the explanation of the dataset that involved the utilization of Tensorflow Hub for ELMo and generating the Bias Word Count from bias lexicon as features to feed into our CNN Classifier that was constructed using Keras with Tensorflow backend for training. The dataset is undergoing data cleaning and pre-processing before feeding into the classifier for training. It is important to note that our validation set might relatively not balanced. This not balance is caused by the random training and validation set split using the scikit-learn module in python.

3.1 Data

The dataset used in this study is the BuzzFeed Webis News Corpus [2]. We retrieved the dataset from Semantic Evaluation (SEMEVAL) International Workshop 2019 portal

[1]. At the point of retrieval of this corpus, there are two separate datasets [18, 19, 20]. The first dataset [1] were gathered from the crowdsourced dataset of 1,273 articles. These articles were political news published in active hyperpartisan and mainstream websites. Three annotators manually labelled each of the articles. The labels annotated were then binarized into hyperpartisan (average 4 or 5) and not (average 1 or 2). Among the 1,273 articles, only 645 articles were chosen as training dataset known as the by-article dataset at the time of retrieval. The remaining is for evaluation. The publishers and articles in the two datasets have no overlap.


```
<?xml version="1.0" encoding="UTF-8"?><articles>
    <article id="0000000" published-at="2017-09-10" title="Kucinich: Reclaiming the money power">
From flickr.com: Money {MID-161793} <p>Money ( <a
    href="https://farm8.static.flickr.com/7020/6551534889_9c8ae52997.jpg" type="external">Image</a> by <a
    href="https://www.flickr.com/people/68751915@N05/" type="external">401(K) 2013</a>) <a
    href="https://creativecommons.org/licenses/by-sa/2.0/" type="external">Permission</a> <a
    type="internal">Details</a> <a type="internal">DMCA</a></p> No Pill Can Stop Tinnitus, But This 1
    Weird Trick Can <p>The walls are closing in on Congress.</p> <p>Terrifying walls of water from
    Hurricanes Harvey and Irma, which, when the damage is totaled, could rise to a half trillion dollars.
    The Walls of War: The multi-trillion dollar ongoing cost of Afghanistan, Iraq and other
    interventions. The crumbling walls of the U.S. infrastructure, which need at least $3 trillion to be
    repaired or replaced. A wall of 11 million undocumented immigrants, whose deportation could easily
    cost $200 billion. The planned wall at the Mexican border, which some estimates place at $67 billion.
    Then there is the Wall of All, the $20 trillion national debt. The walls of debt are closing in.</p>
    <p>At moments of crisis in our nation, in addition to invoking the assistance of Higher powers, we
    can call upon the Constitution for guidance.</p> <p>Article I, Section 8, of the U.S. Constitution
    contains a long-forgotten provision, "the coinage clause," which empowered Congress "to coin (create)
    Money." The ability to create money to meet the needs of the nation is a sovereign power, which
    enables a nation to have control of its own destiny.</p> <p>The same article indicates the Founders
    anticipated having to borrow money on the full faith and credit of the United States. Enter the
    Funding Act of 1790, which assumed and paid off the debt of the colonies and retired the financial
    obligations of the newly created states now united. This was a powerful, object lesson in debt
    retirement, relevant today.</p> <p>It is abundantly clear from a plain reading of the coinage clause
    that the Founders never intended that the only way the government was to be funded was to borrow
    money.</p> <p>The needs of the nation were to come from a system of not borrowing wherein money was a
    neutral value of exchange connecting resources, people and needs, without debt attached.</p> <p>In
```


Fig. 1. Initial raw XML format of the dataset.

The second dataset is called the *by-publisher* dataset. This dataset consists of 754,000 articles. SEMEVAL 2019 task 4 organizer cross-checked two publicly available news publisher bias lists compiled by media professionals from BuzzFeed News and Media Bias Fact Check. 600,000 articles were released as the training dataset, 150,000 articles release as the validation set, while the remaining data were kept private at the time of retrieval.

The dataset was given in Extensible Markup Language (XML) format. The title, article content and the published date is provided. The ground truth of the article is provided in the other file. At this point, the XML parser is required to process the XML and then convert it into Comma Separated Value (CSV) format. This step can be done using the combination of existing python library such as ElementTree.

3.2 Data Cleaning and Pre-Processing

This research was mainly focused on the by-article dataset and does not include any images or videos as input data. Because the articles within the by-article dataset were collected from online news platforms, it contains HTML tags or elements that we need to remove and clean up. The cleaning process started with removing the HTML elements and tags. Then it followed by removing all links (internal and external), hash, and '@' tags.

The cleaning process then checks the advertisement in the data. We have removed the advertisement in the data because the advertisement and browser error messages are noise. The input preserved all the punctuation and other special characters. We split the dataset into training and testing set to avoid compromising the performance of the proposed model. This step can be done using toolkits such as scikit-Learn.

3.3 Implementation and Training

ELMo [3] is a deep contextualized word representation. ELMO developed by AllenNLP in 2018. It uses a deep and bi-directional LSTM model to create word representations. Instead of a dictionary of words and their corresponding vectors, ELMo can analyze words within the text context. Apart from that, this character-based mechanism allowing the model to form representations out of vocabulary words. In short, ELMo representations are contextual. The representations for each word depend on the entire context in which it is used, and the word representations combine all layers of a deep pre-trained neural network. The ELMo vectors assigned to a token are usually a function of the entire sentence containing that word. Unlike traditional word2vec, which gives the same vector for the same word, ELMo allows the same word to exists in different vectors under a different context, an ability to model polysemy [3].

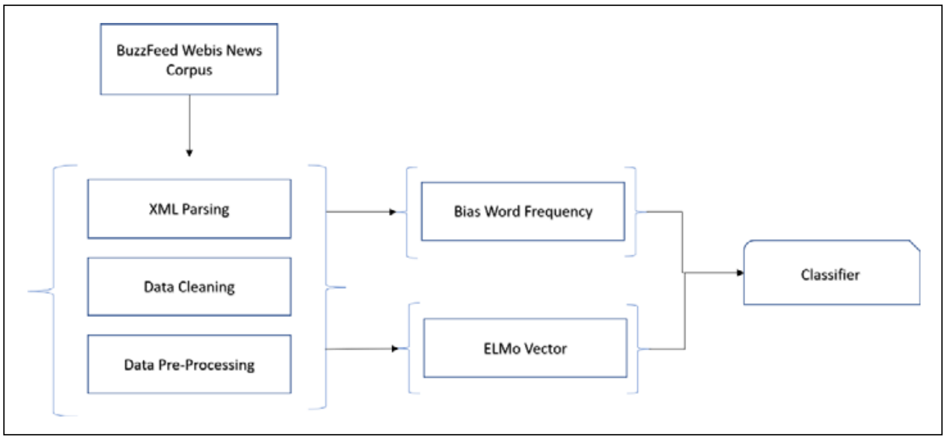

Fig. 2. System pipeline and flow.

For this research, we implement our ELMo using TensorFlow and TensorFlow Hub. We first import the pre-trained ELMo using TensorFlow Hub. Then, we split the dataset into a training set and a validation set. Both the training and validation set are feed into the pre-trained ELMo to generate the ELMo vectors. Refer to this as elmotrain and elmotest, respectively.

We created the CNN Classifier using Keras with Tensorflow backend for the training. The CNN consists of three convolutional layers with kernel sizes (2,3,4). Each layer followed by a Rectified Linear Unit (ReLU) activation function and max-pooling layer. Then, all the output of the max-pooling layers is combined to form the input to the fully connected layer. Since this is a binary classification, a sigmoid activation function is used for the single output at the fully connected layer. The batch size was set to 32, and the epoch used was 50. The loss function used is Binary Cross-Entropy, and the optimizer used is the default Adam optimization algorithm.

We used a bias lexicon to identify bias words from the articles. This research excludes gender bias and only takes into consideration the framing bias. The Bias Lexicon used in this research was a Bias Lexicon built from the NPOV corpus of Wikipedia articles [17]. The authors studied and analyzed the articles marked by the Wikipedia advocates Neutral Point of View policy (NPOV) tag as biased content. Then, the authors construct the NPOV corpus by retrieving all articles that were or had been in the NPOV dispute category together with their full revision history. The authors released the final NPOV data and the bias lexicon and claimed it could be useful in other bias-related tasks. The purpose of using this bias lexicon in this research was to detect the occurrences and the presence of onesided terms, intensifier, or bias-inducing lemmas in the articles. Then, we use the detected occurrences as a feature for the training. In this research, the words in the article that appeared in the bias lexicon are counted. The final frequency of the word occurrence is treated as a feature for the model training at a later stage. A maximum of the first 200 initial tokens per sentence was used as the sentence embedding. A maximum of 200 sentences was used per article as ELMo based vectors with references to the method used by [14, 15]. The elmotrain vectors and bias word frequency feature was used as the input for the classifier. Lastly, perform predictions using the elmotest set.

4. PERFORMANCE EVALUATION

This section describes the accuracy of the proposed method from the previous sections. The result is compared with two baseline models to evaluate the accuracy of the proposed model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

The accuracy is calculated based on Eq. (1), where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives. The classification model implemented in this research used the pre-trained ELMo to generate the embeddings and use the bias word frequency (b) as the features to train a Keras CNN Classifier (KerasCNN) for the classification task. The model then performed prediction using the testing set and able to achieve an accuracy of 73.4%.

Table 1. The accuracy comparison of proposed and baseline models.

	Model	Method	Accuracy
1	Baseline 1	ELMo + CNN + BN	0.822
2	Baseline 2	ELMo + LR	0.608
3	KerasCNN	ELMo & b + CNN	0.734

Although the result of the proposed model still below the best baseline. However, we improved the accuracy of the hyperpartisan classification task. To further investigate the significance and contribution of feature *b* in the training of the Keras CNN model. We had conducted another two investigations. First, the similar features ELMo vectors and bias word frequency were fed into a Logistic Regression (LR) model to test its performance. The LR model was adopted from the Baseline 2 experiment. The prediction on the testing set using the LR model gave a result of 0.692.

Table 2. Accuracy comparison of ELMo + LR model with/without *b* feature.

	Model	Method	Accuracy
1	Baseline 2	ELMo+ LR	0.608
2	LRModel	ELMo & *b* + LR	0.692

The second investigation was conducted to test the performance of the ELMo embeddings. To perform the experiment, the ELMo used to generate the ELMo embeddings was swapped to pre-trained Global Vectors for Word Representation (GloVe). The GloVe + bias (*b*) feature was fed to the same CNN model from the initial experiment (KerasCNN) for training and testing for its performance.

Table 3. Accuracy comparison of GloVe vs. ELMo in CNN with *b* feature.

	Model	Method	Accuracy
1	KerasCNN	ELMo & b + CNN	0.734
2	G-KerasCNN	GloVe & *b* + CNN	0.633

From Table 3, the results demonstrated the strong potential of ELMo in the hyperpartisan classification tasks. Tables 1 and 2 shown that the importance and contribution of bias feature *b* in the classification of the hyperpartisan task. The differences between the word embedding technique such as GloVe, a context-independent word embedding and ELMo, plays a significant role in the natural language processing community. ELMo that can model polysemy gives more advantages as compared to the model that cannot model polysemy. ELMo can produce different vectors for a word based on the linguistic context and for a word with a different meaning under different linguistic contexts. These allow a better representation and cover more linguistic contexts than other word embedding such as GloVe, which combines and summarize the word in different linguistic context into a single vector. In short, GloVe is word context-independent, and ELMo is context-dependent. This research indicates that advanced NLP pre-processing or relevant methods can significantly improve the performance of the ELMo and hyperpartisan tasks.

5. CONCLUSIONS

Hyperpartisan news may harm societies if been manipulated or propagated with ill intention. It is a challenging task to identify and classify hyperpartisan news. In this paper, a method is proposed utilizing ELMo, and bias word frequency as features to identify and classify hyperpartisan news. The results also demonstrated the ability of ELMo and GloVe in identifying and classifying hyperpartisan news. Although it is hard to draw a clear line between "bias" and "unbiased", the framing bias feature generated using the bias lexicon showed an impact in the training of the hyperpartisan classification task. The accuracy result of the proposed method is still below the best baseline result. Therefore, further work is needed to strengthen the bias detection method. This paper only used the *by-articles* dataset for the ELMo embeddings and to train the classifiers. Future work can include the larger by-publisher dataset to improve the hyperpartisan news classification task. This study can also serve as a baseline study to evaluate other hyperpartisan related works.

REFERENCES

1. J. Kiesel, M. Mestre, R. Shukla, E. Vincent, P. Adineh, D. Corney, B. Stein, and M. Potthast, "SemEval2019 Task 4: Hyperpartisan news detection," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, 2019, pp. 829-839.
2. M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff, and B. Stein, "A stylometric inquiry into hyperpartisan and fake news," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 231-240.
3. M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *arXiv Preprint*, 2018, arXiv: 1802.05365.
4. J. Pennington and R. Socher, "GloVe: Global vectors for word representation," in *Proceedings of Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 1532-1543.
5. N. Lee, C. S. Wu, and P. Fung, "Improving large-scale fact-checking using decomposable attention models and lexical tagging," in *Proceedings of Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 1133-1138.
6. Z. Wei, J. Chen, W. Gao, B. Li, L. Zhou, Y. He, and K.-F. Wong, "An empirical study on uncertainty identification in social media context," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 2013, pp. 58-62.
7. Y. Chen, N. J. Conroy, and V. L. Rubin, "News in an online world: The need for an 'automatic crap detector'," in *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, Vol. 52, 2015, pp. 1-4.
8. W. Y. Wang, "'liar, liar pants on fire': A new benchmark dataset for fake news detection," *arXiv Preprint*, 2017, arXiv:1705.00648.
9. A. Madgy and N. Wanas, "Web-based statistical fact checking of textual documents," in *Proceedings of the 2nd International Workshop on Search and Mining User-Generated Contents*, 2010, pp. 103-110.
10. G. L. Ciampaglia, P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini, "Computational fact checking from knowledge networks," *PLoS ONE*, Vol. 10, 2015, No. e0141938.
11. L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. S. H. Wong, and A. Zubiaga, "Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours," in *Proceedings of the 11th International Workshop on Semantic Evaluation*, 2017, pp. 69-76.
12. Y. Long, Q. Lu, R. Xiang, M. Li, and C. R. Huang, "Fake news detection through multiperspective speaker profiles," in *Proceedings of the 8th International Joint Conference on Natural Language Processing*, Vol. 2, 2017, pp. 252-256.
13. B. D. Horne and S. Adali, "This just in: Fake news packs a lot in title, uses simpler,

repetitive content in text body, more similar to satire than real news," in *arXiv Preprint*, 2017, arXiv:1703.09398.
14. Y. Jiang, J. Petrak, X. Song, K. Bontcheva, and D. Maynard, "Team bertha von suttner at SemEval-2019 task 4: Hyperpartisan news detection using ELMo sentence representation convolutional network," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, 2019, pp. 840-844.
15. G. K. W. Huang and J. C. Lee, "Hyperpartisan news and articles detection using BERT and ELMo," in *Proceedings of the 2nd International Conference on Computer and Drone Applications*, 2019, pp. 29-32.
16. J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *arXiv e-Prints*, 2018, arXiv: 1810.04805.
17. M. Recasens, C. Danescu-Niculescu-Mizil, and D. Jurafsky, "Linguistic models for analyzing and detecting biased language," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, Vol. 1, 2013, pp. 1650-1659.
18. T. Isbister and F. Johansson, "Dick-preston and Morbo at SemEval-2019 task 4: Transfer learning for hyperpartisan news detection," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, 2019, pp. 939-943.
19. R. Agerri, "Doris martin at SemEval-2019 Task 4: Hyperpartisan news detection with generic semi-supervised features," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, 2019, pp. 944-948.
20. S. Sengupta and T. Pedersen, "Duluth at SemEval-2019 task 4: The Pioquinto Manterola hyperpartisan news detector," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, 2019, pp. 949-953.

Gerald Ki Wei Huang received the BS degree in Computer Science from Universiti Malaysia Sarawak, Malaysia. He is currently pursuing the MS degree. His research interests include natural language processing and artificial intelligence.

Jun Choi Lee is currently working as a Lecturer at the Faculty of Computer Science and Information Technology in Universiti Malaysia Sarawak. He received his Master degree in Computer Science from Universiti Sains Malaysia in 2008. He had more than ten-year experience in research and development. The area of interest includes natural language processing, data analytics, knowledge engineering and management. He is also a Life Member of Malaysia National Computer Confederation and a registered Graduate Technologist under Malaysia Board of Technologists.