This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

Exploring the Generalisability of Fake News Detection Models

Mr. Nathaniel Hoy
Department of Computer Science
Brunel University
London, United Kingdom
nathaniel.hoy2@brunel.ac.uk

Dr. Theodora Koulouri
Department of Computer Science
Brunel University
London, United Kingdom
theodora.koulouri@brunel.ac.uk

***Abstract*—Fake news has been shown to have a growing negative impact on societies around the world, from influencing elections to spreading misinformation about vaccines. To address this problem, current research has proposed techniques for fake news detection, demonstrating promising results in lab conditions, where models tested on an unseen portion of the same dataset perform well. However, the question of the generalisability of these techniques, and their efficacy in the real-world, is less frequently evaluated. Studies that have looked at generalisability argue that models struggle to distinguish between fake and legitimate news across different topics of news, as well as across different time periods, to the ones on which they have been trained. This prompts the more fundamental question of how well fake news models generalise across news of the same topic and time period. As such, through a series of experiments, this study explores how well popular fake news detection models and features (word-level representations and linguistic cues) generalise across similar news. The first experiment reports high accuracies, when these techniques are tested on an unseen portion of the same dataset, replicating the findings in literature. However, the second experiment reveals that these techniques struggle to generalise well, suffering drops in accuracy of around 50%, when tested against different datasets of the same topic and time period. Exploring possible reasons behind such poor generalisability, the analysis points to the issue of dataset size, motivating the need for larger, more diverse datasets to become available. It also suggests that word-level representations lead to more biased, less generalisable models. Finally, the findings provide preliminary support for the effectiveness of linguistic and stylistic features, and for the potential of features beyond the word or language level, such as URL redirections and reverse image searches.**

***Index Terms*—Fake News Detection, Natural Language Processing, Machine Learning, Generalisability**

I. Introduction

Fake news has been shown to have a significant detrimental effect on societies, such as the misinformation spread during the COVID-19 pandemic and, more recently, the Ukrainian conflict. As such, fake news can be defined as news that is intentionally and verifiably false, typically for the purposes of political influence or profit through advertising [3]. Given the threat that fake news poses, a number of approaches have been developed to address the problem of detecting fake news. The majority of these approaches use textual features to determine whether an article is fake or legitimate. This is achieved through a number of natural language processing (NLP) and machine learning techniques. Typically, the first step involves data pre-processing to clean the text of any data that may introduce noise into the model. This is followed by a feature extraction phase which aims to convert the text into numerical representations or derive statistical features from the text. These numerical representations and features are then fed to a supervised machine learning model, such as a Support Vector Machine (SVM) or a neural network, which is then able to make predictions on other, unseen data [14].

Through a systematic literature review (SLR) conducted by the authors, initial results appear promising with average accuracies of around 80% [11]. However, model evaluation is typically limited to testing with only an unseen portion of the same datasets on which the models were trained, also known as a holdout set. Yet, in order for these models to be considered effective and useful in the real-world, they must be able to be used in a broader context beyond their training dataset. This is known as generalisability. Through the SLR, it was determined that generalisability in fake news can fall into three distinct categories: the first category is the ability to generalise on similar news outside of the dataset that the original model was trained on; for example, a model trained on political news being able to generalise to unseen political news; the second category is the ability to generalise across different news topics, for example, being able to generalise across politics, sport, technology, and celebrity news topics; the third category is generalisability over time; as news is constantly changing and developing, the language used around news also changes over time and this can have an impact on a model trained on older data.

It could be argued that the first test of generalisability is the most fundamental. This is because, if a model cannot generalise on similar but unseen data of the same time period, it seems unlikely to be able to generalise across different topics of news or news from different time periods. As such, the aim of this paper is to explore whether and to what extent fake news detection models can generalise across different datasets of the same topic. The topic on which this study focuses is political news. This paper is organised as follows: first, related literature on methods to detect fake news and generalisability in this area will be discussed. Second, a number of current datasets in this area will be identified and their contents will

This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

be described. Third, the pre-processing steps will be presented before describing the various feature extraction algorithms and machine learning models that will be used in the experiments. Stratified K-fold cross validation will be performed on each combination of feature extraction, machine learning model and dataset before testing these combinations on the remaining datasets. The results will demonstrate how well these models generalise. Finally, the analysis will consider the words that are key in a model in determining how it makes a prediction. The paper will conclude with a discussion of the findings and future work.

II. Related Work

This paper is motivated from a related systematic literature review (SLR) conducted by the authors. The SLR aimed to analyse current approaches to fake news detection and their efficacy, searching a total of 1063 papers between 2016 and 2020 (which is currently being expanded to include 2021 and 2022 papers). The SLR determined that the vast majority of approaches of fake news detection use textual, content-based features and produce relatively promising results of 80% accuracy on average [11]. However, there is little agreement on how best to categorise these textual features. For the purposes of this study, textual features have been broadly divided into two groups:

- **Word-level representations:** which aim to encode the words into numerical vectors with varying degrees of complexity. These include techniques such as Bag-of-Words, TF-IDF and word embeddings [24].
- **Linguistic cues:** that are created through analysis of the corpus text. These may be statistical in nature and include average-word length, sentence complexity, Part-of-Speech (POS) tags, frequency of quotes, pronouns, verbs and named-entities. Some psycholinguistic features may also be derived, such as sentiment analysis scores. Several combinations of features may be created through this analysis [10].

Examples of studies using word-level representations include a study by [13], which explores the use of different feature extraction approaches, such as Bag-of-Words and TF-IDF, in conjunction with various machine learning models such as Naïve Bayes, Logistic Regression, SVMs, Gradient Boosting and Neural Networks. The study tests these models against three different datasets, achieving accuracies as high as 98.3% on a Kaggle dataset. The paper later extends to develop a voting model in an attempt to improve overall accuracy and efficiency. Similarly, [18] also explores the use of Bag-of-Words and TF-IDF across similar models achieving accuracies as high as 92.8% using TF-IDF and an SVM.

Use of linguistic cues can be found in [8], which generated a total of 76 features across 8 different categories which include: "Readability Scores, Linguistic Dimensions, Summative Cues, Affective Cues, Informality Cues, Cognitive Cues, Punctuation Cues, and Time-Orientation Cues". The paper thoroughly examines the use of these groups of features as a combination as well as individually. It reported good performance of 94.2% accuracy on a hold-out set of a single dataset using an SVM and combined features while finding comparative scores of 92.6% accuracy when using only linguistic dimensions with logistic regression.

A combination of word-level representations and linguistic cues may also be used. An example of such approach can be found in [9], which explores the use of statistical features extracted using Grammarly as well as word-embeddings and TF-IDF. Similarly to the previously discussed studies, this approach demonstrated good results when tested on a hold-out portion of a celebrity news dataset of 78% accuracy, as well as a political dataset at 95% accuracy. Taken together, previous research indicates that, regardless of which group or combination of features is chosen, high performances are achievable.

However, unlike the other papers outlined above (and the majority explored in the aforementioned SLR), Gautam and Jerripothula's [9] paper also explores how well these two distinct models generalise across two different news topics: celebrity and political news. The study finds that, for both models, there is a drop in accuracy when testing between the celebrity and political news datasets. The proposed model trained on the political news dataset and tested on celebrity news sees a drop in accuracy of 39% and the proposed model trained on celebrity news dataset and tested on political news sees a drop in accuracy of 8%. While the celebrity model's performance appears to be less affected when testing across datasets, it is important to note that the model only achieved 78% accuracy on its hold-out set, and, therefore, it did not have as much scope to drop as significantly as the political model that achieved 95% accuracy on its hold-out set. Furthermore, both of these datasets are very small in size at 490 articles each and therefore cannot represent reliable generalisability on larger corpora. This finding is also replicated by a study by [5] which also performed a similar test on similar, if not identical, datasets of different topics and found similar drops in accuracy between models trained on political news and tested on celebrity news and vice versa. Similarly, a study by [4] also explores how well models generalise across different topics by testing across two datasets: the ISOT [1], [2] dataset and the Combined Corpus (CC) [25] dataset. Most of the data contained in the ISOT dataset is political in nature whereas the Combined Corpus covers additional topics such as health-care, sports and entertainment. Additionally, these datasets are significantly larger than the datasets used in [9] at 44,898 and 79,548 rows respectively. This experiment therefore is perhaps more representative of generalisability across topics. It is unclear what feature extraction/selection was performed on some of the models. It was found that the hold-out test performance was high at over 90% for each dataset. When testing across datasets, however, a drop in accuracy was observed of approximately 25% on the model trained on the ISOT dataset and tested on the CC dataset. A less significant drop was found between the model trained on the CC dataset and tested on the ISOT dataset of around 15%. This further supports the finding that models do not generalise well across

This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

topics. The less significant drop in accuracy between the CC dataset model and the ISOT dataset could be attributed to the fact that both datasets contain political news whereas the ISOT dataset does not cover all the topics contained in the CC dataset. It is also possible that there is a degree of duplicity between the two datasets as the CC dataset combines data from other datasets which may, in fact, include the ISOT dataset.

While the majority of research report high performing fake news detection models, the above studies suggest that these models do not generalise well across different topics; this gives rise to the more fundamental question of how well models can generalise across different datasets of the same topic. In particular, this study will explore the question of generalisability of fake news detection models across datasets of the same topic (political news) through a series of experiments. First, the datasets that are to be used for this problem will be described as well as the different types of features. This will be followed by building several models, identified in previous literature, and evaluating them to determine if their results are comparable with the ones reported in the literature. These models will then be tested on the remaining datasets that were not used for their training to evaluate how well they generalise.

III. Datasets

There are a number of popular, publicly available datasets that aim to address the fake news problem. As fake news has been defined in several ways and can take different forms, it is important that all the datasets used in this study contain the same type of fake news. As such, this study uses datasets that contain the full text of intentionally false political news articles and do not contain clickbait or satirical political news. It is worth noting that a number of popular datasets are often duplicated under different names. An example of this is the ISOT dataset and the 'Fake and Real News' dataset hosted on Kaggle which request the same citations. Datasets may also contain parts of other datasets. Given the study's focus on generalisability, the datasets selected for this study were crosschecked to reduce the possibility of data duplication.

A. ISOT [1]

The ISOT dataset was created by the Information Security and Object Technology lab at the University of Victoria. It predominantly contains political news from the US and around the world segregated into two CSV files, 'Fake.csv' and 'True.csv'. The fake portion of the dataset contains 23,481 articles collected from unreliable websites as flagged by Politifact.com and Wikipedia. The true portion of the dataset contains 21,417 articles posted on Reuters.com. Both portions of the dataset contain the article title, text, topic and publication date [1], [2].

B. Kaggle Fake or Real [2]

Not to be confused with the 'Fake and Real' Kaggle dataset, this dataset is relatively new according to the publication date on Kaggle, but does not contain any information on the means of collection therefore its reliability may be questionable. Unlike the ISOT dataset, the data is contained in a single CSV file containing the article title, text, and a label of either 'FAKE' or 'TRUE'. The fake and real portions of the dataset are split equally with 3030 articles in each portion of the dataset. In researching datasets for this study, it is interesting to note that this dataset appears identical to the KDNuggets dataset hosted on GitHub.

C. Kaggle Fake News Competition [3]

This dataset was utilised in [12] which performed generalisability study across topics similar to those described in Section II. Similar to the 'Fake or Real' dataset, there is little information regarding its means of collection. In the discussion section of the Kaggle page, the author states that it was collected through combining other datasets hosted on Kaggle. This dataset is also contained in a single CSV file containing the article title, author, text body and label where fake is denoted by the label '0' and true news denoted with the label '1'. The true portion of this dataset contains 10,413 articles and the fake portion contains 10,387 articles.

D. FakeNewsNet [4]

The FakeNewNet dataset is unique in that, unlike other datasets, the authors also provide code which allows you to collect social and spatiotemporal features from Twitter. As none of the other datasets provide such features, for consistency, only the textual features were used from this dataset. There is more than one version of this dataset. For this study, the version containing news from Politifact and Buzzfeed was used. Both contain news regarding US political news from 2016 and were, therefore, combined. This resulted in an overall dataset of 422 articles with 211 labelled as 'fake' and 211 labelled as 'true' [21]–[23].

IV. Methods

The experiments aim to provide a comprehensive evaluation of how common feature extraction methods and supervised learning models perform when tested outside of the datasets on which they have been trained. In effect, the study aims to demonstrate how well these models are likely to perform should they be implemented in the real world.

A. Text Preprocessing

Text preprocessing is an important stage in the NLP pipeline that aims to clean the text of any unwanted noise and allow the resulting models to perform better. In the case of this experiment, a number of steps was carried out to sanitise the data, in particular for Bag of Words and TFIDF approaches. Text was converted to lower-case in order to avoid the subsequent model from treating identical words differently. This was followed by a lemmatization step to further reduce noise by reducing words to their dictionary

[1] https://www.uvic.ca/ecs/ece/isot/datasets/fake-news/index.php
[2] https://www.kaggle.com/datasets/jillanisofttech/fake-or-real-news
[3] https://www.kaggle.com/c/fake-news
[4] https://www.kaggle.com/datasets/mdepak/fakenewsnet

This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

root form. Additionally, remaining punctuation, URLs, Twitter handles, extra whitespace and stop words were also removed as they often provide little additional information and are therefore considered unwanted noise. The exception to this pre-processing was for models utilizing Word2Vec, BERT and linguistic cues. This is because Word2Vec and BERT require contextual information in order to create accurate embeddings and, therefore, only light cleaning was carried out on the input text. This only included converting the input text to lower case, spell checking as well as removing URLs and Twitter handles. Similarly, as linguistic cues require statistical features from the original text, no pre-processing was done when deriving these features.

B. Feature Extraction

A number of different feature extraction approaches relating to word-level representations, such as Bag-of-Words, TF-IDF, word embeddings and linguistic cues were outlined in Section II. In this section, each of these features is discussed in more detail and how they will be implemented as part of this experiment.

1) Bag of Words: is a term frequency-based approach that converts text into a fixed-length vectors by counting how many times each word appears. As this is purely a frequency-based approach, context and word order is not considered. This means that any information on the meaning of the text is lost. In the case of these experiments, SKLearn's Count Vectorizer was utilized to create the Bag-of-Words representations.

2) TF-IDF: extends on the Bag-of-Words approach by using the frequency of a word in a document compared against the frequency of the word across the dataset to determine the importance of the word to the document. This means that very common words achieve a lower score in TF-IDF compared to words that appear less frequently. This allows TF-IDF to capture slightly more meaning about the text than Bag-of-Words, however it also does not consider word order or context. This experiment utilized SKLearn's TFIDF Vectorizer to generate the TF-IDF vectors for each document in the datasets.

3) Word2Vec: attempts to create a large vector space where each word is assigned a vector in the space. These vectors are known as embeddings. Words that are similar will be close in proximity in the vector space and those that are more dissimilar will be further apart. It achieves this by training one of two distinct shallow neural networks: a continuous bag-of-words model (CBOW) or a skip-gram model on a large corpus of text. The CBOW model attempts to predict a target word given the surrounding context words. The skip-gram model attempts to predict the surrounding context words given a single word. Regardless of the model used, the principle is that similar words will have similar context words surrounding them. The weights used to make these predictions therefore must encode the words in such a way that the weights produce similar outputs for similar words. These weights are what are used to generate the vectors for the embeddings [15]. As Word2Vec produces its embeddings based on the surrounding words, this means that Word2Vec encodes more of the meaning of words compared to the previous two approaches. However, similar to the previous approaches, Word2Vec does not consider word-orderings. It is also limited in that it generates the same embedding for words that are morphologically the same but semantically different. An example of this would be the word 'left' which may be used as the past tense 'to leave' or the direction 'left'. This means that Word2Vec is considered context-independent. In these experiments, a pre-trained Word2Vec model trained on a Google News corpus was used to generate the embeddings for the respective datasets.

4) BERT: Similar to Word2Vec, BERT also generates embeddings. However, unlike Word2Vec which produces context-independent embeddings, BERT generates context-dependent embeddings. This means that words that are morphologically the same but semantically different have a unique embedding unlike Word2Vec which produces the same embedding for such words.

A number of key differences allow BERT to produce these unique embeddings. One, is that BERT learns representations at the sub-word level which, unlike Word2Vec, allows BERT to learn representations for words outside of its vocabulary. Additionally, BERT encodes the position of words which also allows the model consider the orderings of words. For BERT to generate meaningful representations, BERT is trained on two tasks: Masked-Language-Modelling (MLM) and Next Sentence Prediction (NSP). In MLM, a word in a given sequence is masked and the goal of the model is to accurately predict the masked word by using the words either side of the masked word to make the prediction. This allows BERT to learn the relationship between words. In NSP, pairs of sentences are given to BERT whereby it attempts to determine if sentence B comes after sentence A. This allows BERT to learn the relationship between sentences. Through the use of Transformers, these tasks can be trained in parallel with a massive amount of data relatively quickly [16]. Similar to Word2Vec, we use a pre-trained BERT model, in this case the BERT base-uncased model, to generate the embeddings.

5) Linguistic Cues: As discussed in Section II, linguistic cues are typically statistical in nature or derived from the text, such as with sentiment analysis. This experiment uses the set of 34 linguistic features ('Linguistic Dimensions' and 'Punctuation Cues') which was identified as producing the best performance in fake news classification in a series of experiments by [8]. In these experiments, after these features were collected, they were combined to form a 34-dimensional vector that was then used for training on each model. A summary of the features collected from each document in the respective datasets is presented in Table 1.

C. Local Interpretable Model-Agnostic Explanations (LIME)

The LIME package aims to make it easier to understand the classifications that the black-box models used in machine learning make. For the purposes of this study, the submodule 'LimeTextExplainer' was used, which, given an

This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

TABLE I
ENGINEERED LINGUISTIC FEATURES

Feature	Description
Word Count	Total Number of Words
Syllables Count	Total number of syllables
Sentence Count	Total number of sentences
Word/Sent	Average number of words per sentence
Long Words Count	Number of words with more than 6 characters
All Caps Count	Number of words in all caps
Unique Words Count	Number of unique words
Personal Pronouns %	% of words such as 'I, we, she, him'
First Person Singular %	% of words such as 'I, me'
First Person Plural %	% of words such as 'we, us'
Second Person %	% of words such as 'you, your'
Third Person Singular %	% of words such as 'she, he, her, him'
Impersonal Pronouns %	% of words such as 'it, that, anything'
Articles %	& of words such as 'a, an, the'
Prepositions %	% of words such as 'below, all, much'
Auxiliary Verbs %	% of words such as 'have, did, are'
Common Adverbs %	% of words such as 'just, usually, even'
Conjunctions %	% of words such as 'until, so, and, but'
Negations %	% of words such as 'no, never, not'
Common Verbs %	% of words such as 'run, walk, swim'
Common Adjectives %	% of words such as 'better, greater, larger'
Concrete Figures %	% of words that represent real numbers
Punctuation Count	Total number of punctuation marks
Full Stop Count	Total number of full-stops
Commas Count	Total number of commas
Colons Count	Total number of colons
Semi-Colons Count	Total number of semi-colons
Question Marks Count	Total number of question marks
Exclamation Marks Count	Total number of exclamation marks count
Dashes Count	Total number of dashes
Apostrophe Count	Total number of apostrophes
Brackets Count	Total number of brackets

SKLearn pipeline and any given text, returns an array of tuples containing a word and a number indicating whether the word had an impact in the model classifying one way or another (in the case of binary classification). Within the scope of this study, words that carry a negative score mean they contributed to a 'fake' classification. Words that carry a positive score mean they contributed to a 'real' classification.

V. RESULTS

The following experiments aim to demonstrate how well different supervised learning models trained using different groups of features generalise across the four political news datasets outlined in Section III. The first experiment aims to validate the performance of popular techniques presented in literature by training and testing them on the same dataset. Using this performance as the baseline, the second experiment evaluates the generalisability of these models and feature sets trained on one dataset by testing them on the remaining three datasets.

A. Baseline Stratified K-Fold Cross Validation

In the case of models that utilised Bag-of-Words and TF-IDF, text was pre-processed using the steps outlined in Section IV to remove any noise. Word2Vec and BERT were excluded from this step as they require the surrounding words to derive their embeddings. Likewise, linguistic cues were also omitted from this step as stop-words and punctuation formed a large portion of the collected features.

Initially, a stratified k-fold cross-validation (SCV) test was performed to provide a baseline to compare against for all combinations of features, ML models and datasets. SCV was chosen to ensure that both the train and test set had an even class distribution to avoid overfitting, in particular for the smaller datasets such as FakeNewsNet. A total of k=5 folds were chosen thus representing an 80/20 split between the training data and the test data for each iteration. Furthermore, by splitting the datasets into 5 folds, we are able to gain a more representative set of evaluation metrics averaged across each split compared to a traditional hold-out test. The results from this test are presented in Tables II-V.

The results demonstrate comparable performances to the ones reported in the literature, in particular for the ISOT, Kaggle Fake or Real and Kaggle Competition datasets. In general, all models across all feature extraction approaches perform well when tested on the same dataset, achieving performances between 86% and 94% accuracy along with comparable precision and recall. The FakeNewsNet dataset sees poor results, but this is likely due to its small size and the fact that only textual features were used in this experiment and did not collect any of the social features that are available and have typically been used with this dataset in the literature. The SVMs trained on linguistic cues also see poorer performance compared to other types of machine learning model. This may be because the model hyperparameters were kept the same across all different feature types. Likewise, the neural network trained on linguistic features sees slightly worse performance across datasets. This may be due to the fact that the early stopping hyperparameter was used to prevent the model from overfitting when training on word-level representations and the model may not have had enough epochs to train for linguistic cues.

B. Cross-Dataset Testing

In order to determine how well models generalise across similar datasets, each model trained in Section V is tested with each dataset it was not trained on. This involved using the same instance of the feature extraction method used to create the original model to transform the text from the other datasets. The same instance of the models was also used to make predictions on the other datasets. This was to ensure that feature-sets and decision boundaries did not change when testing across datasets.

Due to the volume of models being tested in this manner, and the relatively low amount of variance between baseline models as shown in Tables II-V, three sets of analysis were performed. The first analysis, shown in Table VI, demonstrates the average performance of models trained on one dataset followed by average performance when testing these models on each of the remaining datasets. The second analysis, shown in Figure I, demonstrates the average performance of all models trained on one dataset against the average performance

This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

TABLE II
FAKENEWSNET SCV RESULTS

Feature	Model	Acc	Pre	Rec	F1
Count	AdaBoost	0.53	0.53	0.53	0.53
	Gradient Boosting	0.55	0.55	0.55	0.54
	Logistic Regression	0.56	0.56	0.56	0.55
	Neural Network	0.50	0.49	0.56	0.48
	Random Forest	0.54	0.54	0.54	0.53
	SVM	0.50	0.35	0.50	0.39
TF-IDF	AdaBoost	0.53	0.53	0.53	0.52
	Gradient Boosting	0.54	0.55	0.54	0.53
	Logistic Regression	0.56	0.56	0.56	0.55
	Neural Network	0.51	0.52	0.67	0.57
	Random Forest	0.57	0.57	0.57	0.56
	SVM	0.50	0.35	0.50	0.36
Word2Vec	AdaBoost	0.56	0.56	0.56	0.55
	Gradient Boosting	0.57	0.57	0.57	0.56
	Logistic Regression	0.60	0.61	0.60	0.60
	Neural Network	0.51	0.49	0.59	0.51
	Random Forest	0.53	0.53	0.53	0.52
	SVM	0.50	0.35	0.50	0.39
BERT	AdaBoost	0.56	0.57	0.56	0.56
	Gradient Boosting	0.60	0.61	0.60	0.59
	Logistic Regression	0.59	0.59	0.59	0.59
	Neural Network	0.54	0.54	0.53	0.53
	Random Forest	0.55	0.55	0.55	0.55
	SVM	0.50	0.35	0.50	0.39
Linguistic Cues	AdaBoost	0.53	0.53	0.53	0.52
	Gradient Boosting	0.56	0.57	0.56	0.56
	Logistic Regression	0.59	0.59	0.59	0.58
	Neural Network	0.53	0.50	0.60	0.53
	Random Forest	0.52	0.52	0.52	0.51
	SVM	0.50	0.25	0.50	0.33

TABLE III
KAGGLE COMP SCV RESULTS

Feature	Model	Acc	Pre	Rec	F1
Count	AdaBoost	0.93	0.93	0.93	0.93
	Gradient Boosting	0.94	0.94	0.94	0.94
	Logistic Regression	0.95	0.95	0.95	0.95
	Neural Network	0.96	0.96	0.96	0.96
	Random Forest	0.88	0.89	0.87	0.88
	SVM	0.89	0.88	0.89	0.89
TF-IDF	AdaBoost	0.93	0.93	0.93	0.93
	Gradient Boosting	0.94	0.93	0.94	0.94
	Logistic Regression	0.95	0.95	0.95	0.95
	Neural Network	0.96	0.96	0.95	0.95
	Random Forest	0.88	0.89	0.87	0.88
	SVM	0.94	0.94	0.93	0.94
Word2Vec	AdaBoost	0.84	0.84	0.84	0.84
	Gradient Boosting	0.87	0.87	0.87	0.87
	Logistic Regression	0.88	0.88	0.87	0.87
	Neural Network	0.87	0.85	0.75	0.80
	Random Forest	0.83	0.84	0.82	0.82
	SVM	0.88	0.88	0.87	0.87
BERT	AdaBoost	0.86	0.86	0.85	0.86
	Gradient Boosting	0.90	0.90	0.89	0.89
	Logistic Regression	0.94	0.94	0.94	0.94
	Neural Network	0.92	0.90	0.88	0.89
	Random Forest	0.84	0.85	0.82	0.83
	SVM	0.90	0.90	0.89	0.89
Linguistic Cues	AdaBoost	0.97	0.97	0.96	0.97
	Gradient Boosting	0.97	0.97	0.97	0.97
	Logistic Regression	0.95	0.96	0.95	0.95
	Neural Network	0.95	0.85	0.86	0.85
	Random Forest	0.97	0.97	0.96	0.97
	SVM	0.61	0.79	0.55	0.45

TABLE IV
KAGGLE FAKE OR REAL SCV RESULTS

Feature	Model	Acc	Pre	Rec	F1
Count	AdaBoost	0.86	0.87	0.86	0.86
	Gradient Boosting	0.89	0.89	0.89	0.89
	Logistic Regression	0.92	0.92	0.92	0.92
	Neural Network	0.93	0.94	0.93	0.94
	Random Forest	0.84	0.85	0.84	0.84
	SVM	0.85	0.87	0.85	0.85
TF-IDF	AdaBoost	0.86	0.86	0.86	0.86
	Gradient Boosting	0.90	0.90	0.90	0.90
	Logistic Regression	0.91	0.92	0.91	0.91
	Neural Network	0.93	0.93	0.93	0.93
	Random Forest	0.83	0.84	0.83	0.83
	SVM	0.90	0.91	0.90	0.90
Word2Vec	AdaBoost	0.85	0.85	0.85	0.85
	Gradient Boosting	0.89	0.89	0.89	0.89
	Logistic Regression	0.87	0.87	0.87	0.87
	Neural Network	0.86	0.80	0.85	0.82
	Random Forest	0.84	0.85	0.84	0.84
	SVM	0.89	0.89	0.89	0.89
BERT	AdaBoost	0.86	0.86	0.86	0.86
	Gradient Boosting	0.89	0.89	0.89	0.89
	Logistic Regression	0.91	0.91	0.91	0.91
	Neural Network	0.90	0.88	0.89	0.88
	Random Forest	0.83	0.84	0.83	0.83
	SVM	0.88	0.88	0.88	0.88
Linguistic Cues	AdaBoost	0.82	0.82	0.82	0.82
	Gradient Boosting	0.84	0.84	0.84	0.84
	Logistic Regression	0.81	0.81	0.81	0.81
	Neural Network	0.79	0.72	0.71	0.71
	Random Forest	0.83	0.83	0.83	0.83
	SVM	0.54	0.72	0.54	0.42

TABLE V
ISOT SCV RESULTS

Feature	Model	Acc	Pre	Rec	F1
Count	AdaBoost	0.99	0.99	0.99	0.99
	Gradient Boosting	0.99	0.99	0.99	0.99
	Logistic Regression	0.99	0.99	0.99	0.99
	Neural Network	0.99	0.99	0.99	0.99
	Random Forest	0.98	0.98	0.97	0.98
	SVM	0.90	0.91	0.91	0.90
TF-IDF	AdaBoost	0.99	0.99	0.99	0.99
	Gradient Boosting	0.99	0.99	0.99	0.99
	Logistic Regression	0.98	0.98	0.98	0.98
	Neural Network	0.99	0.99	0.99	0.99
	Random Forest	0.97	0.97	0.97	0.97
	SVM	0.96	0.96	0.96	0.96
Word2Vec	AdaBoost	0.94	0.94	0.93	0.93
	Gradient Boosting	0.95	0.95	0.95	0.95
	Logistic Regression	0.96	0.96	0.96	0.96
	Neural Network	0.97	0.95	0.96	0.95
	Random Forest	0.94	0.94	0.94	0.94
	SVM	0.92	0.92	0.92	0.92
BERT	AdaBoost	0.96	0.96	0.96	0.96
	Gradient Boosting	0.97	0.97	0.97	0.97
	Logistic Regression	0.99	0.99	0.99	0.99
	Neural Network	0.98	0.98	0.98	0.98
	Random Forest	0.96	0.96	0.96	0.96
	SVM	0.95	0.95	0.95	0.95
Linguistic Cues	AdaBoost	0.95	0.95	0.94	0.94
	Gradient Boosting	0.94	0.94	0.94	0.94
	Logistic Regression	0.90	0.90	0.90	0.90
	Neural Network	0.89	0.84	0.88	0.86
	Random Forest	0.94	0.94	0.94	0.94
	SVM	0.51	0.70	0.53	0.39

This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

across all remaining datasets broken down by each kind of model discussed in Section IV. The third, shown in Figure II, shows a similar break down but by feature group.

TABLE VI
BASELINE AVERAGE ACCURACY & CROSS-DATASET AVERAGE ACCURACY COMPARISON

Dataset	Baseline SCV Avg. Acc.	Cross-Dataset Avg. Acc.
ISOT	0.95	0.52
Kaggle FoR	0.86	0.52
Kaggle Comp	0.9	0.36
FakeNewsNet	0.54	0.54

Table VI aims to demonstrate how well models generalise depending on the dataset on which they are trained. It shows the average performance of all models and features for each dataset (Baseline SCV Avg. Acc.) compared to the average performance of each of these models tested on the remaining datasets. It is clear from this comparison that, regardless of the dataset used for training, models suffer a dramatic drop in accuracy across the remaining datasets, with the worst drop being observed for the models trained on the Kaggle Competition models (53% accuracy drop), while the accuracy of models trained on the Kaggle Fake or Real and the ISOT dataset dropped by 37% and 42% respectively. As the FakeNewsNet models only achieved performance that mirrors random classification when tested on an unseen portion of its own dataset, it is not surprising that the performance is equally poor when testing it on the remaining datasets.

C. Generalisability by Model

Further analysis aimed to determine whether any of the machine learning models used in testing generalise better than other machine learning models for each dataset. Similar to Table VI, Figure I shows the poor generalisability of these models, where the original baseline model accuracy is relatively high, dropping when the model is applied to the other datasets. This analysis failed to indicate that the choice of model relates to better generalisability.

D. Generalisability by Feature Type

Next, the analysis focuses on whether any of the features discussed in Section IV generalise better than other groups of features across all datasets. Similar to Figure I, Figure II shows the poor generalisability of these approaches. Of the word-level representations (Bag-of-Word, TF-IDF, Word2Vec and BERT), there appears to be no representation of words that outperforms any of the others in terms of generalisability. Similarly, linguistic cues are not found to generalise well across datasets. However, the performance of linguistic cues appears to suffer significantly less compared to the performance of all other types of features, when tested on the Kaggle Competition dataset. Linguistic cues also appear to perform more conistently in terms of their generalisability across datasets which may suggest they are less sensitive to changes in dataset.

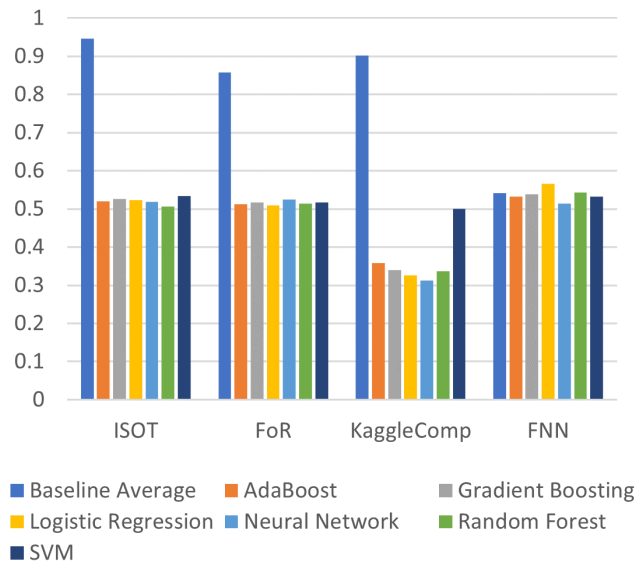

Fig. 1. Cross-Dataset Performance by ML Model

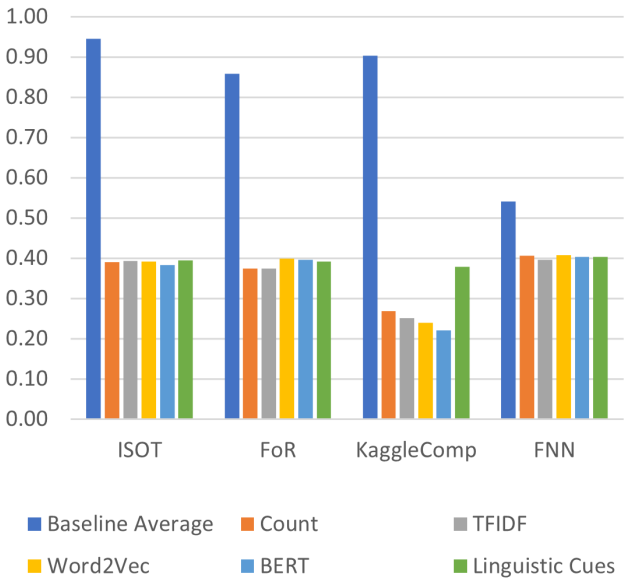

Fig. 2. Cross-Dataset Performance by Feature Type

E. Interpreting Models Trained on Word-Level Representations

The previous sections demonstrated the poor generalisability of machine learning models trained on a variety of word-level representations and linguistic cues. To further understand the reasons behind the poor generalisability on word-level representations, the LIME package was used, as detailed in Section IV. The LimeTextExplainer submodule was used to generate two lists of words: a list of words that were most fundamental to classifying a document as ‘fake’, and a list of words that were most fundamental to classifying a document as ‘real’ [19]. As LIME relies on an SKLearn pipeline that

This article has been accepted for publication in a future issue of this conference proceedings, but has not been fully edited. Content may change prior to final publication. Citation information: DOI10.1109/BigData55660.2022.10020583, 2022 IEEE International Conference on Big Data (Big Data)

TABLE VII
FREQUENCY DISTRIBUTION OF KEYWORDS CONTRIBUTING TO CLASSIFICATION

ISOT				Kaggle FoR				Kaggle Comp			
Real		Fake		Real		Fake		Real		Fake	
Word	Freq	Word	Freq	Word	Freq	Word	Freq	Word	Freq	Word	Freq
Reuters	52	trump	32	president	20	2016	17	Clinton	14	Mr	26
Washington	19	just	20	state	19	Hillary	13	Hillary	11	president	20
Wednesday	13	image	11	Obama	13	October	13	2016	11	Ms	16
Trumps	11	Obama	11	house	12	election	11	October	8	twitter	15
Tuesday	11	Hillary	9	told	10	Russia	7	war	8	follow	11
minister	11	don	8	says	10	FBI	6	share	8	com	7
house	6	like	8	sanders	9	article	6	election	5	united	7
Friday	6	Could	7	campaign	8	just	6	Obama	4	new	6
Government	5	GOP	6	white	8	email	5	LA	4	news	5
Thursday	5	Doesn't	6	debate	8	war	5	source	4	Breitbart	5
election	5	Black	5	republican	7	world	4	Aleppo	4	Sunday	5
court	4	Americans	4	senate	7	Comey	3	November	4	percent	5
EU	4	Right	4	voters	6	share	3	FBI	4	York	4
month	4	Video	4	Islamic	6	daily	3	UK	3	Trumps	4

includes an SKLearn vectorizer and machine learning model, a Logistic Regression model was trained using TF-IDF features as part of the pipeline, similar to the methods employed by [7]. Training was performed for each dataset and lists were generated using unseen documents from each dataset. Due to the computational complexity of LIME, this analysis used 100 unseen documents from each dataset. Next, a frequency distribution of the top 15 keywords that increase the likelihood of a model classifying 'real' or 'fake' for each dataset was produced. The FakeNewsNet dataset was excluded from this analysis, because of its extremely poor baseline performance, as identified by the experiment detailed in Section V. Table VII shows the ranked list of 15 keywords that contributed to the classification for each dataset.

The first pair of columns in Table VII shows the frequency of key words used by the models trained on the ISOT dataset. Keywords contributing to a 'real' classification include the word 'Reuters'. This is unsurprising as Reuters is often considered a reliable news website and was used as part of the collection for the 'real' portion of the ISOT dataset, as discussed in Section III. Conversely, the 'fake' column shows keywords such as 'Trump' and 'Obama' as well as 'Hillary'. This suggests that the articles from the dataset focused on the topic of the 2016 presidential election . This is further supported by the fact the term 'election' appears in the 'real' column.

Similar to the ISOT columns, the terms in the Kaggle Fake or Real columns, suggest that articles in this dataset also surround the 2016 Presidential Election. This is due to keywords used in classifying a document as 'fake' such as 'Hillary', 'election', '2016' and 'October'. The 'real' column shows fewer keywords indicating this, but some can be seen such as 'Sanders', 'campaign' and 'candidates'.

Also similar to the previous two datasets, the two columns corresponding to the Kaggle Competition dataset indicate that it also focuses on the 2016 presidential election, using keywords for classification that include '2016', 'Hillary', 'October' 'Clinton' and 'election'. Interestingly, compared to the ISOT and Fake or Real datasets, these keywords appear in the in the 'real' column as opposed to the 'fake' column. This may explain why models trained on this dataset suffered the highest drop in accuracy when tested on the other datasets. Keywords for 'fake' classification appear not to focus on any distinctive topic however the word 'Breitbart' gives some indication that part of the 'fake' portion of this dataset was collected from Breitbart.com – a website sometimes argued to be spreading misinformation.

Looking at each pair of columns across the three datasets explored, it is evident that at all the datasets have a strong focus on the 2016 presidential election. This demonstrates that the data was collected around the same time period. Furthermore, the majority of terms that exist across all datasets confirm that the datasets focus on political news. As the goal of this study is to test generalisability across similar datasets of the same topic and time period, these observations provide confidence in the study's selection of datasets, which would have been a legitimate concern given the limited metadata available for each one as discussed in Section III.

VI. DISCUSSION

This study was motivated following an SLR carried out by the authors, which indicated that generalisability is seldom addressed in the literature when developing and evaluating methods for fake news detection [11]. The empirical study described in this paper sets out to explore how well existing fake news detection techniques generalise across different datasets in the same news domain (political news) and time period. In the first experiment of this study, a set of models was developed and tested using Stratified K-Fold Cross-Validation. It was found that the models trained and tested on the same dataset produced high performances, which were comparable to, and replicated, results reported in the literature.

In the second experiment (detailed in Section V), these models were tested on the remaining datasets to determine how well they generalise to different datasets of the same topic and time period. The core finding of this analysis is that current techniques of fake news detection do not generalise

well to similar news. This raises questions around the efficacy of current techniques in the real-world, given that models must be able to perform outside of the datasets on which they are trained. This finding motivates the need for more robust models. It also adds to the body of evidence that suggests that current publicly available datasets are simply too small to train generalisable models [4]. This may explain why models trained on the ISOT dataset performed marginally better in terms of generalisability compared to models trained on the smaller datasets, and it further demonstrates the need for larger, well-labelled datasets, such as those hosted by Facebook, to be made more readily available.

Furthermore, this experiment demonstrated that word-level representations (BoW, TFIDF, Word2Vec and the state-of-the-art BERT) are the worst performing in terms of generalisability, regardless of the type of model that is used in training, as detailed in Section V. However, linguistic cues led to better and more consistent performance in comparison to word-level representations. This finding is in line with [5] who reported positive results for generalisability over-time and across domains, using linguistic features. Further evidence supporting the argument that linguistic cues can perform better in terms of generalisability can be found in [9], [12]. A possible explanation is that linguistic cues do not rely on the content of the article, but rather on stylistic features, to determine whether an article is fake or not. Another factor contributing to poor generalisability may be also be the words that are used to derive word-level representations, where words such as named-entities are deemed important by the model but only serve to bias the model to classify based on such words. Removing such words that increase bias in datasets has been found to improve generalisability as reported by a recent study by [26]. Moreover, the use of novel statistical techniques, such as causality analysis, that give more weighting to words that tend to generalise better could also improve model generalisability [17].

The final stage of the analysis provided support to the bias argument, aiming to identify the words which had the largest impact on classifying a document as 'fake' or 'real'. Inspection of the frequency distributions revealed that, in several cases, the 'fake' and 'real' word lists contain words that are very specific to the datasets on which they are trained. That is, the same words that are influential to classify an article as 'fake' in one dataset are associated with real news in another dataset. For example, words relating to Hillary Clinton occur frequently in the 'fake' word list for the Kaggle Fake or Real dataset but also in the 'real' word list of the Kaggle Competition dataset. Similarly, the word 'Reuters' occurred 52 times out of the 100 documents explored in the ISOT frequency distribution, suggesting that the model considered any article containing this term to be 'true'. This may explain why models trained on these datasets perform poorly in terms of generalisability, as particular words are expected to be related to each class of fake or real news, producing biased models. It could be argued that such words relating to the source should be removed however, in the current iteration of the SLR preliminary findings suggest that this is seldom done (hence, why they were included for the purposes of this study to evaluate the performance of current approaches in terms of generalisability). This provides additional evidence to motivate the need to remove such words, or select words more intelligently, when training models using word-level representations, in the pursuit of more generalisable models in future research. This finding expands on [5] who argued that the frequency of certain words on each side of the dataset may have a negative impact on generalisability, and also aligns well with a study by [7] which drew similar conclusions using LIME on South African news. Taken together, these findings suggest that techniques relying only on word-level representations may not support generalisable models despite being the most popular approach in the broader literature.

The analysis suggested that the linguistic cues selected for this study performed marginally better compared to word-level representations. However, their overall generalisability is still lacking. The study used a set of 34 linguistic cues, which previous literature has found to produce better outcomes. More selective, finer-grained experiments could pinpoint which of these 34 linguistic cues, or combinations of cues, are the most essential and effective for fake news detection. Moreover, it could be argued that, given the right combination of linguistic cues, additional novel features, as well as intelligently selected word-level representations, good generalisability may be achieved for datasets of the same domain. Examples of novel features include frequency of URL redirections [6], volume of advertising (as profit for advertising is often a motivation for producing fake news [3]) and reverse image search to determine if images have been manipulated or used out of context [20]. As such, exploring such combinations of features with the view of improving generalisability should be the primary focus of future research.

VII. Future Work & Conclusions

This study tested the generalisability of a number of supervised machine learning models trained on a variety of word-level representations and linguistic cues across four datasets. The study demonstrates that word-level representations do not generalise well on current, popular, publicly available datasets of the same topic, in this instance, the 2016 US Presidential election. This is regardless of the ML model used. Through the LIME package, the paper suggests that word-level representations may have a negative impact on generalisability due to dataset bias, possibly as a result of the limited size of such datasets. As such, linguistic cues appear to be more robust than word-level representations in terms of generalisability, but due to the number of possible combinations of these features and little exploration into generalisability in literature, it is not clear what features would generalise best.

As such, this study makes a number of recommendations for future research. Primarily, that future work in this area utilise generalisability tests as part of the evaluation process. This is to ensure that resulting models are robust and usable in a real-world context. Additionally, that the datasets used are of

a more significant size to aid in increasing the generalisability of resulting models and ideally contain the URLs of articles, such that a broader range of novel features may be gathered. A larger dataset may also contribute to avoiding the bias observed in current datasets demonstrated by this study. Furthermore, future studies should look to explore different combinations of linguistic features as well as derive new, novel features that can be used for fake news detection, such as those suggested in Section VI. This should be to determine whether these new features generalise better than those explored in this article and the wider literature.

REFERENCES

[1] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10618 LNCS:127–138, 2017.

[2] Hadeer Ahmed, Issa Traore, and Sherif Saad. Detecting opinion spams and fake news using text classification. *Security and Privacy*, 1(1):e9, Jan 2018.

[3] Hunt Allcott and Matthew Gentzkow. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 31(2):211–236, May 2017.

[4] Ciara Blackledge and Amir Atapour-Abarghouei. Transforming Fake News: Robust Generalisable News Classification Using Transformers. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 3960–3968. IEEE, Dec 2021.

[5] Sonia Castelo, Thais Almeida, Anas Elghafari, Aécio Santos, Kien Pham, Eduardo Nakamura, and Juliana Freire. A Topic-Agnostic Approach for Identifying Fake News Pages. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, pages 975–980, New York, NY, USA, 2019. Association for Computing Machinery.

[6] Zhouhan Chen and Juliana Freire. Discovering and Measuring Malicious URL Redirection Campaigns from Fake News Domains. *Proceedings - 2021 IEEE Symposium on Security and Privacy Workshops, SPW 2021*, pages 1–6, May 2021.

[7] Harm de Wet and Vukosi Marivate. Is it Fake? News Disinformation Detection on South African News Websites. In *2021 IEEE AFRICON*, volume 2021-Septe, pages 1–6. IEEE, Sep 2021.

[8] Aaron Carl T. Fernandez and Madhavi Devaraj. Computing the Linguistic-Based Cues of Fake News in the Philippines Towards its Detection. In *Proceedings of the 9th International Conference on Web Intelligence, Mining and Semantics - WIMS2019*, pages 1–9, New York, New York, USA, 2019. ACM Press.

[9] Akansha Gautam and Koteswar Rao Jerripothula. SGG: Spinbot, Grammarly and GloVe based Fake News Detection. In *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*, pages 174–182. IEEE, Sep 2020.

[10] Georgios Gravanis, Athena Vakali, Konstantinos Diamantaras, and Panagiotis Karadais. Behind the cues: A benchmarking study for fake news detection. *Expert Systems with Applications*, 128:201–213, Aug 2019.

[11] Nathaniel Hoy and Theodora Koulouri. A Systematic Review on the Detection of Fake News Articles. Oct 2021.

[12] Maria Janicka, Maria Pszona, and Aleksander Wawer. Cross-Domain Failures of Fake News Detection. *Computación y Sistemas*, 23(3):1089–1097, Oct 2019.

[13] Sawinder Kaur, Parteek Kumar, and Ponnurangam Kumaraguru. Automating fake news detection system using multi-level voting model. *Soft Computing*, 24(12):9049–9069, Jun 2020.

[14] Kamran Kowsari, Kiana Jafari Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura E. Barnes, and Donald E. Brown. Text Classification Algorithms: A Survey. *Information 2019, Vol. 10, Page 150*, 10(4):150, Apr 2019.

[15] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, Jan 2013.

[16] Britney Muller. BERT 101: State Of The Art NLP Model Explained. Available at: https://huggingface.co/blog/bert-101 (2022/11/20).

[17] Bo Ni, Zhichun Guo, Jianing Li, and Meng Jiang. Improving Generalizability of Fake News Detection Methods using Propensity Score Matching. Jan 2020.

[18] Karishnu Poddar, Geraldine Bessie Amali D., and K.S. Umadevi. Comparison of Various Machine Learning Models for Accurate Detection of Fake News. In *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, volume 1, pages 1–5. IEEE, Mar 2019.

[19] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *NAACL-HLT 2016 - 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Demonstrations Session*, pages 97–101, Feb 2016.

[20] Diego Saez-Trumper. Fake tweet buster. In *Proceedings of the 25th ACM conference on Hypertext and social media*, pages 316–317, New York, NY, USA, Sep 2014. ACM.

[21] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News on Social Media. *Big Data*, 8(3):171–188, Jun 2020.

[22] Kai Shu, Suhang Wang, and Huan Liu. Exploiting Tri-Relationship for Fake News Detection. 2017.

[23] Kai Shu, Suhang Wang, and Huan Liu. Beyond News Contents: The Role of Social Context for Fake News Detection. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, pages 312–320, New York, NY, USA, 2019. Association for Computing Machinery.

[24] Aswini Thota, Priyanka Tilak, Simrat Ahluwalia, Nibrat Lohia, Simeratjeet Ahluwalia, and Nibhrat Lohia. Fake News Detection: A Deep Learning Approach. *SMU Data Science Review*, 1(3):10, 2018.

[25] Junaed Younus Khan, Md Tawkat, Islam Khondaker, Sadia Afroz, Gias Uddin, and Anindya Iqbal. A Benchmark Study of Machine Learning Models for Online Fake News Detection. 2021.

[26] Yongchun Zhu, Qiang Sheng, Juan Cao, Shuokai Li, Danding Wang, and Fuzhen Zhuang. Generalizing to the Future. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, volume 1, pages 2120–2125, New York, NY, USA, Jul 2022. ACM.