Explainable Misinformation Detection from Text: A Critical Look

Suzana Bašić, Marcio Fuckner, and Pascal Wiggers

Amsterdam University of Applied Sciences
{s.basic, m.fuckner, p.wiggers}@hva.nl

Abstract. With the proliferation of misinformation on the web, automatic methods for detecting misinformation are becoming an increasingly important subject of study. If automatic misinformation detection is applied in a real-world setting, it is necessary to validate the methods being used. Large language models (LLMs) have produced the best results among text-based methods. However, fine-tuning such a model requires a significant amount of training data, which has led to the automatic creation of large-scale misinformation detection datasets. In this paper, we explore the biases present in one such dataset for misinformation detection in English, NELA-GT-2019. We find that models are at least partly learning the stylistic and other features of different news sources rather than the features of unreliable news. Furthermore, we use SHAP to interpret the outputs of a fine-tuned LLM and validate the explanation method using our inherently interpretable baseline. We critically analyze the suitability of SHAP for text applications by comparing the outputs of SHAP to the most important features from our logistic regression models.

Keywords: misinformation detection · dataset bias · LLM · XAI · SHAP.

1 Introduction

The increase of misinformation on the web is recognised as a socially relevant issue and acknowledged by several authors ([23, 10, 29, 31, 2]). To mitigate the risks of exposing unreliable content, many initiatives took place to check the content's reliability, either manually or automatically. Manual checking could lead to reliable results using experts with access to external sources. However, this task comes at a price of low scalability, limiting checking to a small subset of news articles.

A plethora of techniques has been proposed to automate the verification of the integrity of the news. The main approaches encompass propagation-based and content-based methods, as well as combinations thereof. Propagation-based methods use network features, i.e. features that encode information about how news spreads on social networks. On the other hand, content-based methods use the linguistic features of the text of the article and possibly images in the case of multimodal methods. This encompasses a wide variety of methods, from traditional machine learning models using hand-engineered features or bag-of-words

(BOW) representations to neural networks with non-contextual word embeddings and, most recently, transfer learning with large language models (LLMs). As with most natural language processing (NLP) tasks, LLMs reportedly achieve the best results among the content-based methods.

Fine-tuning such models requires a significant amount of training data, which can be found in various large-scale unreliable news datasets [11, 19, 14, 12, 7, 8]. Since labelling a large dataset requires considerable time and expertise, large datasets are increasingly being created semi-automatically, which can cause problems with data quality. Therefore, the question is how well models trained on such data generalise in real-world settings. The results of our experiments on NELA-GT-2019 [7] show that models are at least partly learning the stylistic and other features of different news sources rather than the actual features of unreliable news. We observed a considerable reduction in model performance on unseen data when using training and test sets with no news site overlap as opposed to randomly selected sets. In addition, we observed that a simple baseline achieved comparable accuracy results to Transformer models when using non-biased data. We therefore suggest that a further investigation of potential improvements to the inherently interpretable baselines could lead to more sustainable and less resource-intensive procedures.

Apart from the ever increasing resource requirements of large language models, a major concern is their lack of explainability. That is especially relevant in real-world settings, where automatic methods are increasingly being applied to flag web content or limit its reach. In our view, such automated actions should be accompanied by some form of explanation so as to increase transparency and user trust. Since LLMs are not inherently interpretable, the dominant approach is to use a model-agnostic post-hoc explanation method, such as SHAP [9]. However, it is unclear to what degree SHAP-based explanations reflect the actual workings of language models. We found the NELA-GT dataset family to be the perfect testing ground for an exploratory investigation of that question because the datasets — and, consequently, the models trained on them — contain very specific biases, which were observed in previous work and confirmed by our own experiments. We therefore apply SHAP to interpret the outputs of a language model fine-tuned on NELA-GT-2019 and validate the explanation method using our inherently interpretable baseline. The main contribution of our work is a comparative analysis of feature importance between logistic regression and SHAP, which illuminates certain shortcomings of SHAP explanations for text applications.

2 Related Work

This section lists the most prominent datasets and methods used for automatic misinformation detection.

2.1 Misinformation Detection Datasets

A lot of attention has been dedicated in recent years to improving misinformation detection performance. The task of training effective models depends on high-quality datasets. This section lists the most prominent misinformation detection datasets in English.

LIAR [26] is a fake news detection dataset containing 12.8k human-annotated short statements collected from the fact-checking website PolitiFact. Statements are annotated using six fine-grained labels. Apart from the statements, the dataset contains metadata about the speaker and the statement's context. **Fakeddit** [11] is a multimodal dataset with over one million samples automatically collected from Reddit. The data samples include text, images, comments data, and metadata. The data is labelled using 2-way, 3-way, and 6-way labels, enabling both fine-grained and coarse-grained classification. **BuzzFace** [19] consists of over 1.6 million Facebook comments discussing 2,282 news articles. The articles were annotated by BuzzFeed using four labels. The dataset includes additional metadata.

The **NELA-GT** datasets [12, 7, 8] are large collections of news articles scraped from the web during 2018, 2019, and 2020. News outlets are labeled as *reliable*, *mixed* or *unreliable* based on the information from several fact-checking organisations. Individual articles are assigned the corresponding label automatically based on the site-level labels. This work focuses on the NELA-GT family of datasets as we are interested in misinformation in longer texts.

2.2 Misinformation Detection Methods

There is a rapidly growing literature on automatic misinformation detection. **Propagation-based methods** [23, 10, 29] use data from social networks, including data about the individual users who share and comment on news, as well as broader news sharing patterns in the network. **Content-based methods** use features based on the content of an article. Multimodal content-based methods use a combination of text and images contained in a piece of news [27, 30, 2]. Purely text-based methods are also frequently used [1, 14, 4]. However, it is not entirely clear what kind of linguistic features are most useful. For example, Bozarth and Budak [3] speculate that models based on engineered linguistic features are more robust, while Gravanis et al. [6] have found that they add little to the performance of word embeddings. Surprisingly, Zhou et al. [31] observe that BERT only slightly outperforms BOW on r/Fakeddit. **Mixed methods** use a combination of propagation-based and content-based features [17, 2].

In the area of **explainable misinformation detection**, few inherently explainable systems have been developed, e.g. dEFEND [21]. Yang et al. [28] construct an explainable fake news detector that uses text and metadata. Reis et al. [15] conduct a large-scale exploration of linguistic and network features used for explainable misinformation detection. Since state-of-the-art NLP models are not inherently explainable, post-hoc explanation methods are typically used to explain the predictions of purely content-based models. These methods, like LIME

[16] and SHAP [9], determine feature importance for individual prediction by observing the relationship between changes in the input to the model and the model's output.

Zhou et al. [31] highlight the importance of collecting high-quality datasets and suggests improvements to data collection, dataset construction and experiment design processes in order to avoid hidden pitfalls that lead to biased models. Since unreliable news detection is generally a classification task, datasets should have a pair of article features and annotated labels. Labels can be collected in different ways, and the level of effort may vary. For simplicity, we are taking into account only the outcome and not the process of assigning these labels. We can divide labels into two types: article-level and site-level labels. Article-level labels are fine-grained and found in smaller datasets because maintaining such datasets requires significant time and expertise. In contrast, site-level annotations are scalable since articles from the same news outlet receive the same label, favouring scalability but compromising performance.

Another concern regarding data collection pertains to biased resources. For example, FakeNewsNet [22] uses Google search to query the original news article. It comes with the price of potentially selecting the wrong article due to the ranking process. NELA datasets, on the other hand, collect news directly from outlets, avoiding such risk. Also, selection bias can be generated by fact-checking websites that link unreliable classes mostly with articles with click-bait titles containing celebrity names and similar. Bozarth and Budak [3] alert that models that used the random permutation approach to split train and test presented biased behaviour, bypassing the actual task of classifying reliable and unreliable news and just memorising the site identities or writing style of some news sources, even when the names of news sources are removed from the data.

3 Experimental Setup

This section describes the design choices we made to conduct the experiments in terms of data collection, model construction, and evaluation.

3.1 Data

We use a subset of the NELA-GT-2019 [7] dataset for misinformation detection in news articles. The entire 2019 dataset consists of 1.12M news articles published in 2019 and scraped from 260 different news sources. Each news source is labelled as "reliable", "mixed" or "unreliable" based on the reliability scores aggregated from seven fact-checking websites. Rather than containing article-level labels, the dataset contains site-level labels, meaning that all articles from the same news source receive the same label. This is a clear limitation because a predominantly unreliable news source can occasionally publish reliable news and vice versa.

For our experiments, we use two different subsets of NELA-GT-2019. Each subset consists of 10k training articles, 3k validation articles and 3k test articles. Preprocessing included removal of very short texts ($<$ 150 chars) and

reducing the size of large texts to a maximum of 500 tokens. Preliminary experiments showed that a ten-fold increase in data size did not significantly influence model performance. We exclude the "mixed" category and balance the number of "reliable" and "unreliable" samples in each of the data splits. The two subsets are constructed in the same way as in Zhou et al. [31] Namely, in the first ("random") subset, articles are randomly split between the training, validation and test data splits. In the second ("disjoint") subset, we ensure that the news sources are strictly separated across the data splits, so that no articles from the same source appear both in the training and test data. This enables us to test the performance of our models on articles from previously unseen news sources.

3.2 Models

As a baseline model we use a logistic regression classifier on unigram, bigram and trigram features encoded using a TF-IDF vectorizer with a maximum vocabulary size of 30k. We purposefully do not exclude features that appear in a large number of documents because, as with news outlet names, preliminary experiments showed that they are useful for our feature importance analysis. We do, however, exclude features with low frequency in the entire dataset, retaining only the 30k most frequent features. The training and development sets were merged before training this model because, rather than using a separate development set, ten-fold cross-validation is used for hyperparameter search. Input documents are constructed by concatenating the title and article strings.

Furthermore, we fine-tune DistilBERT [18] for sequence classification. DistilBERT belongs to a family of large-scale pre-trained language models which have become popular in recent years thanks to their state-of-the art performance on many standard natural language processing tasks. It is a smaller, faster version of BERT [5], a language representation model which learns deep bidirectional (sub-)word embeddings using a Transformer encoder [25]. Compared to BERT, DistilBERT is reported to retain 97% of language understanding capabilities with a 40% smaller size and 60% more speed [18]. The optimal dropout values were obtained empirically and modified as follows: attention_dropout=0.2, dropout=0.2, seq_classif_dropout=0.3. As input to the model, we use the title and text of an article separated by the special token [SEP].

3.3 Explanation methods

SHAP (SHapley Additive exPlanations) [9] is a model-agnostic post-hoc explanation method for explaining individual model predictions. It is based on Shapley values [20, 24], a method from cooperative game theory for assigning payouts to players based on their individual contributions to the total payout in the game. The SHAP framework encompasses model-specific variants of the explanation method for tree-based, linear, and deep models, as well as a kernel-based estimation method that connects Shapley values with LIME [16]. SHAP determines feature contributions by perturbing the values of input features and observing

the effects on model output. We use SHAP to analyze the individual predictions of our fine-tuned Transformer models.

To assess the reliability of the method, we also apply SHAP to the baseline models. Since our subsets of the data are not too large, it was possible to calculate SHAP values for the simple baseline models on the entire dataset. That means that we were able to construct a global feature importance overview, aggregated from the individual feature importance of each prediction in the dataset. Thanks to this, we were able to analyze the global feature importance as constructed by SHAP, as well as compare it to the feature importance based on the logistic regression model coefficients.

4 Results

This section shows the results of our experiments, focused on the effect of the chosen dataset split strategy on model performance. A brief overview of results in terms of model explanations is given, but feature importance is analyzed in more detail in Section 5.

Table 1: Logistic regression results on the random data split

	Precision	Recall	F1-score	No. of documents
0	0.8482	0.8220	0.8349	3000
1	0.8273	0.8530	0.8399	3000
Accuracy	0.8375	0.8375	0.8375	6000
Macro avg.	0.8378	0.8375	0.8374	6000

Table 2: Logistic regression results on the disjoint data split

	Precision	Recall	F1-score	No. of documents
0	0.7331	0.7700	0.7510	3000
1	0.7578	0.7196	0.7382	3000
Accuracy	0.7448	0.7448	0.7448	6000
Macro avg.	0.7454	0.7448	0.7446	6000

4.1 Model performance

The performance of the models is presented in Tables 1, 2, 3, and 4. The classes 0 and 1 represent reliable and unreliable news, respectively.

Our baseline model achieves an accuracy of 83.75% on the random data split and 74.48% on the disjoint split. DistilBERT performs significantly better on

Table 3: DistilBert results on the random data split

	Precision	Recall	F1-score	No. of documents
0	0.9116	0.9150	0.9133	3000
1	0.9146	0.9113	0.9130	3000
Accuracy	0.9131	0.9131	0.9131	6000
Macro avg.	0.9131	0.9131	0.9131	6000

Table 4: DistilBert results on the disjoint data split

	Precision	Recall	F1-score	No. of documents
0	0.7119	0.8533	0.7762	3000
1	0.8169	0.6546	0.7268	3000
Accuracy	0.7540	0.7540	0.7540	6000
Macro avg.	0.7644	0.7540	0.7515	6000

the random split, achieving 91.32%. However, on the disjoint split it performs comparably to the baseline, achieving an accuracy of 75.40%. Precision and recall are mostly balanced, except for the DistilBERT model on the disjoint data split.

4.2 Model Explanations

This section presents the most important features of the logistic regression model as determined by both the model weights and SHAP. The analysis of the results in terms of feature importance is limited to logistic regression for two reasons. Firstly, we do not have access to an internal feature importance ranking of DistilBERT as we do for logistic regression. That means that we are unable to compare the results of SHAP to a straightforward "gold standard" of feature importance as in the case of logistic regression. Secondly, calculating SHAP values is computationally expensive. Therefore, it is impractical to run SHAP on the entire dataset to construct a global feature importance overview. We focus on the logistic regression model trained on the randomly split portion of the dataset, but similar effects can also be observed on the disjoint data split. The global ordering of features by SHAP values is built by averaging the absolute instance-level SHAP values for each feature.

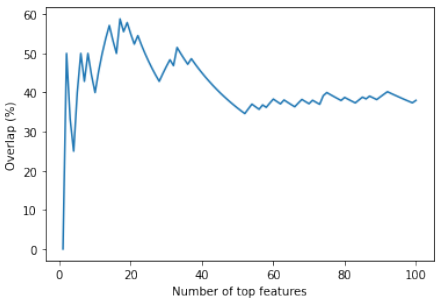


Fig. 1: The percentage of overlap between logistic regression and SHAP (y-axis) among n most important features identified by the two methods (x-axis) on the random data split

Table 5: The top 50 most important features from the logistic regression model and SHAP on the random data split. The columns LR and SHAP contain features that are only identified as being in the top 50 by either logistic regression or SHAP, respectively. The middle two columns list the features that are found in both logistic regression and SHAP, ordered by the feature importance in the two models.

LogReg	**Intersection (LogReg order)**	**Intersection (SHAP order)**	**SHAP**
ap	apos	apos	during
natural news	read more	said	us
com	tass	but	in
stated	read	mr	it
illegal	mr	this	had
article	but	video	on
mr trump	said	read	at
cnsnews	says	says	and
cnsnews com	video	read more	has
democrat	reported	media	year
2019 at	according	according	was
ms	this	according to	first
natural	president trump	reported	trump
breitbart	according to	said the	america
centre	obama	president trump	war
this article	however	tass	its
buy new	media	obama	the
article was	said the	however	israel
vaccine			not
the democrats			say
this article was			even
music			epstein
msnbc			any
moscow			president
cnn			that
tass the			more than
trump donald john			or
rt			is
below			the us
schiff			that the
percent			an
donald john			than

Figure 1 illustrates the percentage of overlap between the top features learned by logistic regression and identified by SHAP. The overlap is higher than 50% only for certain values between 13 and 33, with the highest value of 58.82% for the 17 most important features. When the number of features is higher than 46, the percentage of overlap stabilises between 35 and 40%.

Table 5 shows the 50 most important features as identified by the logistic regression model and SHAP. The two outer columns, *LogReg* and *SHAP*, contain the features found among the 50 most important features only by the respective method. They are ordered by their importance rank, from the most important to the least important feature. The inner two columns, *Intersection (LogReg order)* and *Intersection (SHAP order)* contain the features found in the top 50 features by both methods. The only difference between those two columns is that the first one is ordered by the feature importance according to the logistic regression model, while the second one reflects the feature importance according to SHAP.

5 Discussion

In this section we discuss the results presented in Section 4 and dive deeper into the comparison of prediction explanations obtained from the logistic regression model and SHAP. Furthermore, we present the observations from a qualitative analysis of SHAP explanations of DistilBERT predictions.

5.1 Model accuracy

On the portion of the dataset with no news site overlaps, the accuracy of our baseline decreases by 9.27 percentage points, while the accuracy of DistilBERT decreases by 15.92 points. This indicates that a bias towards learning the features of news sources is present in both models. The DistilBERT models suffers a significantly larger decrease in accuracy, indicating that the more complex LLM is more biased than the baseline. These findings are in line with those of Zhou et al. [31], who looked at the generalisability of different models on the related NELA-GT-2018 dataset.

Apart from the need for data debiasing techniques already mentioned by Zhou et al. [31], these findings also point to potentially interesting considerations regarding model choice. While LLMs are very powerful models that achieve state-of-the-art results on most standard natural language processing tasks, they are not without flaws. They require significant data, time, and computing power to train and fine-tune, and their outputs are difficult to interpret. If a simple baseline achieves comparable results on reasonably non-biased data, it might be worth exploring potential improvements to the inherently interpretable baselines rather than using a more resource intensive model.

5.2 Logistic regression explanations

As can be seen from the results presented in Table 5, SHAP performs rather poorly in detecting the features that are relevant to the model. By inspecting the top features obtained from the logistic regression model and SHAP, we can observe two interesting effects.

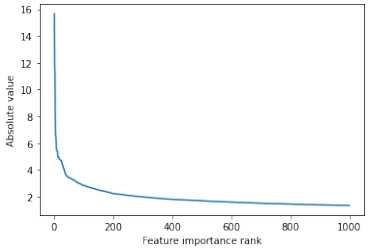
Firstly, SHAP fails to identify nearly all news outlet names found among the 50 most important features learned by logistic regression. The outlet names not identified by SHAP, with their respective feature importance rank, are as follows: **ap** (7), **natural news** (9), **cnsnews** (23), **cnsnews com** (24), **breitbart** (30), **msnbc** (41), **cnn** (43), and **rt** (46). The only news outlet name that SHAP recognised is **tass**, but it was ranked much lower by SHAP than by logistic regression (the 30th vs 3rd most important feature). Furthermore, the bigram **tass the** additionally appears as the 44th most important feature in logistic regression, but not in SHAP. That means that SHAP fails to capture the source bias, which is present both in the data and the model, as confirmed by our data manipulation experiments. This might point to a general shortcoming of SHAP regarding bias detection in textual datasets.

The second effect that can be observed is that most of the top features that are identified only by SHAP do not carry much semantic content. In linguistics, only nouns, main verbs, adjectives, and adverbs are typically considered as content words. Word classes such as determiners, conjunctions, and prepositions, are considered to be function words, which primarily or exclusively carry grammatical rather than lexical meaning. While such features are typically excluded from BOW models based on word frequency or a predefined list of stop words, we intentionally include them in order to assess the faithfulness of SHAP explanations to the classification model. In fact, 20 out of the 32 features exclusive to SHAP are function words or combinations thereof. Those features and their respective ranking in the logistic regression feature importance are: **in** (179), **it** (116), **had** (67), **on** (114), **at** (86), **and** (462), **has** (139), **was** (557), **its** (358), **the** (6557), **not** (344), **even** (66), **any** (131), **that** (1125), **more than** (52), **or** (268), **is** (906), **that the** (247), **an** (353), and **than** (202). Most of the listed features are placed well below the rank of 50 by logistic regression. While the ranks of 200 or 300 might seem close to 50 in a space of 30,000 features in total, it is important to note that the distribution of feature contributions has a very long tail, as shown in Figure 2. Figure 2a gives a broader overview of the top 1000 feature contributions, while Figure 2b zooms in on the 100 most important features. It is clear that the features below the top 50 are far less important in the model.

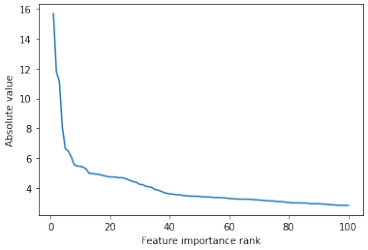
5.3 DistilBERT explanations

Since attempting to construct global SHAP explanations for DistilBERT would require subsampling the dataset due to computational complexity, we instead discuss the observations based on a qualitative inspection of the instance-level explanations of DistilBERT predictions. While we cannot make any conclusions about the global feature ranking, we have observed some patterns in the several dozen explanations we analysed.

As with the explanations of logistic regression predictions, function words seem to be rather prominent in the top 20 features for individual instances. This is illustrated in Figure 3a. The highlighted words and phrases include **is it on**, **and**, **and each and**, **and then after that I**, **somewhat**, and similar. In this

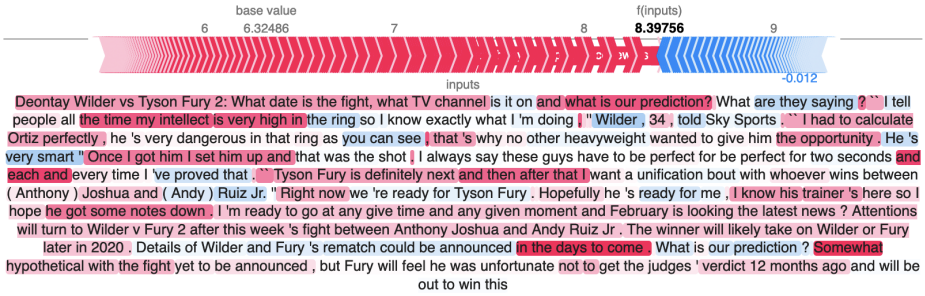


(a) Feature importance of the top 1000 features

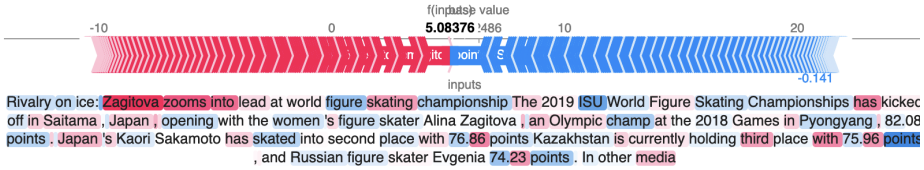


(b) Feature importance of the top 100 features

Fig. 2: The long tail of feature contributions. Figs 2a and 2b show the absolute feature contributions for the top 1000 and 100 most important features



(a) SHAP explanation of an instance-level DistilBERT prediction, illustrating the prominence of function words in the explanation



(b) SHAP explanation of an instance-level DistilBERT prediction, illustrating an inconsistency in the explanation

Fig. 3: Two examples of SHAP explanations: Features are highlighted in different colours: red for those that contribute to unreliable news and blue for reliable news.

example, those sequences consist exclusively of function words. However, that is not always the case, especially with longer highlights. If an entire sentence is highlighted, one might intuitively expect that the content words have the highest SHAP values in the sentence, but that is not necessarily the case. That means that text highlights, which are used specifically to explain the output of

BERT-like models, do not always clearly reflect the actual feature contributions determined by SHAP.

Apart from the function words, we can also see that individual punctuation marks are highlighted in this example. More specifically, 10 out of the top 20 features for this text are punctuation marks. As in the case of function words, this is not always as visible as in this example because punctuation marks are usually highlighted as a part of a longer phrase or sentence. We have also observed punctuation marks among the top features in other examples, but not as frequently as function words. Moreover, they seem to be much more frequent in examples from the test set on which the model makes an incorrect prediction. Therefore, the prominence of punctuation marks might indicate uncertainty of the predictive model on those examples. This effect was not observed with the BOW model because punctuation is excluded automatically by scikit-learn's [13] TfidfVectorizer.

Figure 3b illustrates a different issue, which we have observed in various examples. The article begins with the title: *Rivalry on ice: Zagitova zooms into lead at world figure skating championship*. In the title, the word **skating** is highlighted red, signifying that it contributes to the unreliable class. However, in the first sentence it is highlighted blue, as contributing to the reliable class. Similarly, the name Zagitova is highlighted in dark red in the title, signifying a significant contribution to the unreliable class. In contrast, it is not highlighted in the first sentence at all. Even though the vector representations of the same word differ across contexts and the features are not independent, we consider it unlikely that these two features would drastically impact the model prediction in these two very similar contexts (the title and first sentence). While we did not perform an extensive analysis of instance-level explanations for the logistic regression model, we did notice a similar issue. In several instances, the same word was listed as the top contributing feature for both the reliable and unreliable class across different examples, which is impossible. That indicates an issue with the reliability of SHAP explanations.

6 Conclusion and Future Work

In this paper, we investigated how data collection approaches can directly affect the performance of models in realistic scenarios. Results indicated that models that apparently outperformed the baseline had a significant drop in performance when the news outlet name was used as a basis to create disjoint sets for training and testing. Furthermore, we conducted an extensive analysis of SHAP explanations of our models, highlighting potential shortcomings of the approach. The main contributions of this paper and potential future research directions are as follows.

Data bias. We found that models trained on NELA-GT datasets learn certain features of news sources rather than the features of unreliable news. This confirms earlier findings on bias in the NELA-GT datasets [e.g. 31]. We showed

that the same effects are still observed on an iteration of the dataset containing a significantly larger variety of news sources (260 sources in NELA-GT-2019 compared to 194 in NELA-GT-2018). That problem stems from the data collection method, where all articles from the same news source are automatically assigned the same label based on the rating of the news source by fact-checking organisations. This process results in a relatively unclean dataset containing an unknown number of mislabeled articles as well as biases based on news outlet names and on the lexical and stylistic choices of particular authors and/or outlets.

Model performance. We found that a simple baseline almost matched the performance of a state-of-the-art Transformer model on a debiased portion of the dataset. Based on this, we recommend further investigation into the effects of data bias on comparative model performance. While LLMS achieve state-of-the-art results on a vast array of NLP tasks, we consider it worthwile to explore under which conditions a smaller, faster, more sustainable, and more interpretable model might achieve similar performance.

Validity of SHAP explanations for text. We discovered great inconsistencies between the most important features of our interpretable model and SHAP explanations of the same model. Our analysis showed that SHAP does not identify the news source biases present in the data and models, while it does highlight a large number of irrelevant features. We consider this our most important contribution because post-hoc explanations of black-box models are often taken for granted and they cannot be validated directly. We conclude that SHAP might not be a suitable method for textual data. In future work, we aim at the development of text-specific explanation methods or extensions of existing post-hoc explanation methods for text.

Bibliography

[1] Ahmed, H., Traore, I., Saad, S.: Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques. In: Traore, I., Woungang, I., Awad, A. (eds.) Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments, vol. 10618, pp. 127–138. Springer International Publishing, Cham (2017), series Title: Lecture Notes in Computer Science

[2] Ajao, O., Bhowmik, D., Zargari, S.: Fake News Identification on Twitter with Hybrid CNN and RNN Models. In: Proceedings of the 9th International Conference on Social Media and Society. pp. 226–230. ACM, Copenhagen Denmark (Jul 2018)

[3] Bozarth, L., Budak, C.: Toward a Better Performance Evaluation Framework for Fake News Classification. Proceedings of the International AAAI Conference on Weblogs and Social Media **14**(1) (May 2020)

[4] Conforti, C., Pilehvar, M.T., Collier, N.: Towards Automatic Fake News Detection: Cross-Level Stance Detection in News Articles. In: Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). pp. 40–49. Association for Computational Linguistics, Brussels, Belgium (2018)

[5] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 4171–4186 (2019)

[6] Gravanis, G., Vakali, A., Diamantaras, K., Karadais, P.: Behind the cues: A benchmarking study for fake news detection. Expert Systems with Applications **128**, 201–213 (Aug 2019)

[7] Gruppi, M., Horne, B.D., Adalı, S.: NELA-GT-2019: A Large Multi-Labelled News Dataset for The Study of Misinformation in News Articles. arXiv:2003.08444 [cs] (Mar 2020), arXiv: 2003.08444

[8] Gruppi, M., Horne, B.D., Adalı, S.: NELA-GT-2020: A Large Multi-Labelled News Dataset for The Study of Misinformation in News Articles. arXiv:2102.04567 [cs] (Feb 2021), arXiv: 2102.04567

[9] Lundberg, S.M., Lee, S.I.: A Unified Approach to Interpreting Model Predictions. In: Advances in neural information processing systems. vol. 30 (2017)

[10] Monti, F., Frasca, F., Eynard, D., Mannion, D., Bronstein, M.M.: Fake News Detection on Social Media using Geometric Deep Learning. arXiv:1902.06673 [cs, stat] (Feb 2019), arXiv: 1902.06673

[11] Nakamura, K., Levy, S., Wang, W.Y.: r/Fakeddit: A New Multimodal Benchmark Dataset for Fine-grained Fake News Detection. In: Proceedings of the 12th Language Resources and Evaluation Conference. pp. 6149–6157 (2020)

[12] Norregaard, J., Horne, B.D., Adali, S.: NELA-GT-2018: A Large Multi-Labelled News Dataset for The Study of Misinformation in News Articles. In: Proceedings of the international AAAI conference on web and social media. vol. 13, pp. 630–638 (2019)

[13] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D.: Scikit-learn: Machine Learning in Python. The Journal of Machine Learning Research **12**, 2825–2830 (2011)

[14] Rashkin, H., Choi, E., Jang, J.Y., Volkova, S., Choi, Y.: 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. pp. 2931–2937. Association for Computational Linguistics, Copenhagen, Denmark (2017)

[15] Reis, J.C.S., Correia, A., Murai, F., Veloso, A., Benevenuto, F.: Explainable Machine Learning for Fake News Detection. In: Proceedings of the 10th ACM Conference on Web Science - WebSci '19. pp. 17–26. ACM Press, Boston, Massachusetts, USA (2019)

[16] Ribeiro, M., Singh, S., Guestrin, C.: "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 1135–1144 (2016)

[17] Ruchansky, N., Seo, S., Liu, Y.: CSI: A Hybrid Deep Model for Fake News Detection. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. pp. 797–806. ACM, Singapore Singapore (Nov 2017)

[18] Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108 [cs] (Feb 2020), arXiv: 1910.01108

[19] Santia, G.C., Williams, J.R.: BuzzFace: A News Veracity Dataset with Facebook User Commentary and Egos. In: Twelfth international AAAI conference on web and social media (2018)

[20] Shapley, L.S.: A value for n-person games, Contributions to the Theory of Games, 2, 307–317. Princeton University Press, Princeton, NJ, USA (1953)

[21] Shu, K., Cui, L., Wang, S., Lee, D., Liu, H.: dEFEND: Explainable Fake News Detection. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 395–405. ACM, Anchorage AK USA (Jul 2019)

[22] Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H.: 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)

[23] Shu, K., Wang, S., Liu, H.: 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. pp. 312–320. ACM, Melbourne VIC Australia (Jan 2019)

[24] Strumbelj, E., Kononenko, I.: Explaining prediction models and individual predictions with feature contributions. Knowledge and Information Systems **41**(3), 647–665 (2014)
[25] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, , Polosukhin, I.: Attention is All you Need. In: Advances in neural information processing systems. vol. 30 (2017)
[26] Wang, W.Y.: "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 422–426 (2017)
[27] Wang, Y., Ma, F., Jin, Z., Yuan, Y., Xun, G., Jha, K., Su, L., Gao, J.: EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. In: Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining. pp. 849–857 (2018)
[28] Yang, F., Pentyala, S.K., Mohseni, S., Du, M., Yuan, H., Linder, R., Ragan, E.D., Ji, S., Hu, X.: XFake: Explainable Fake News Detector with Visualizations. The World Wide Web Conference on - WWW '19 pp. 3600–3604 (2019)
[29] Yang, S., Shu, K., Wang, S., Gu, R., Wu, F., Liu, H.: Unsupervised Fake News Detection on Social Media: A Generative Approach. Proceedings of the AAAI Conference on Artificial Intelligence **33**, 5644–5651 (Jul 2019)
[30] Yang, Y., Zheng, L., Zhang, J., Cui, Q., Li, Z., Yu, P.S.: TI-CNN: Convolutional Neural Networks for Fake News Detection. arXiv:1806.00749 [cs] (Jun 2018), arXiv: 1806.00749
[31] Zhou, X., Elfardy, H., Christodoulopoulos, C., Butler, T., Bansal, M.: Hidden Biases in Unreliable News Detection Datasets. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. pp. 2482–2492 (2021)