360 Degrees Rumor Detection: When Explanations Got Some Explaining To Do

Bram Janssens[a,b,c,*] , Lisa Schetgen[a,*], Matthias Bogaert [a,b], Matthijs Meire[d], Dirk Van den Poel [a,b]

[a]Ghent University, Department of Marketing, Innovation, and Organization, Tweekerkenstraat 2, 9000 Ghent, Belgium
[b] FlandersMake@UGent–corelab CVAMO
[c] Research Foundation Flanders
[d] IESEG School of Management, Univ. Lille, CNRS, UMR 9221 - LEM - Lille Economie Management, 3 Rue de la Digue, Lille F-59000, France
[*] Both authors contributed equally
Bram.Janssens@UGent.Be, Lisa.Schetgen@UGent.Be, Matthias.Bogaert@UGent.Be (Corresponding author), M.Meire@Ieseg.Fr, Dirk.VandenPoel@UGent.Be

Abstract
Unverified rumor detection recently received considerable academic attention due to the societal impact resulting from this potential misinformation. Previous work in this area mainly focused on textual features using a limited number of data sets and candidate algorithms, and completely disregarded model explainability. This study aims to come up with a more comprehensive social media rumor detection methodology. First, we investigate which machine or deep learning algorithm is best suited to classify tweets into rumors and non-rumors using both textual and structured features. Next, we interpret these rumor detection models with the LIME method and assess the quality of the explanations via fidelity and stability. To ensure the robustness of our methodology, it is benchmarked across the well-known PHEME data sets and two novel data sets, which are made publicly available. The results indicate that machine learners perform best on small data sets, while transformer architectures show the highest predictive accuracy for larger data sets. Unfortunately, these high accuracy transformer models are incompatible with LIME, which results in low fidelity. Moreover, our study shows that all LIME explanations are unstable across folds. Based on these results, we argue to evaluate explanation quality using fidelity and stability before explanation deployment. Our results further demonstrate that apparent model-agnostic explanations such as LIME do not seem to be completely model-agnostic and should be used with caution.

Keywords: Analytics, Explainable Artificial Intelligence, Explanation Quality Evaluation, Rumor Detection, Text mining

1. Introduction

Over the past years, social media has been increasingly adopted as an instant communication channel by businesses and news outlets (Stieglitz et al., 2014). For example, the passing of Queen Elizabeth II was first announced on Twitter (Theil, 2022), which showcases the platform's rising importance in direct information sharing. Especially during a crisis, people are using social media as one of their primary sources of information (Cinelli et al., 2020). Unfortunately, the use of social media

implies that information can be easily shared and spread without proper restriction or verification. In this context, a rumor is defined as a piece of text that is unverified at the moment of sharing and spread throughout the network of social media users (Bondielli & Marcelloni, 2019). During the COVID-19 pandemic, the spread of rumors posed threats to the democratic decision making and fed the anxiety of the public opinion (Reisach, 2021).

Given this societal impact, rumor analysis has become an important field of research which can be divided into three subtasks: rumor detection or identification, rumor veracity classification, and rumor stance classification (Bondielli & Marcelloni, 2019). This study focuses on rumor detection which is a binary classification problem to predict whether a piece of text contains unverified information (i.e., a rumor) or not (i.e., non-rumor). Given the importance of identifying rumors as soon as possible, existing research has been mainly devoted to improving the performance of automatic rumor detection techniques (Al-Seram et al., 2019). The used techniques can be broadly categorized into machine learning (ML) and deep learning (DL) models. The former apply intensive feature engineering to extract useful information from both the text and context of a social media post and feed these features into traditional machine learning models, such as logistic regression and random forest (Yang et al., 2015). The latter form a specific subgroup of machine learning models, which have a neural network-based architecture with multiple hidden layers. These hidden layers allow the model to uncover complex non-linear relationships and extract different underlying data representations (Zinovyeva et al., 2020). Deep learning methods have become increasingly popular for textual data classification since they learn the required numeric vector representation automatically in the training phase, thereby skipping the extensive feature engineering process (Al-Sarem et al. 2019).

Whereas research has made great improvements in successfully identifying rumors, we identify three main gaps in the current rumor detection literature. First, studies implementing deep learning models only use textual data as input and implement novel architectures to improve predictive accuracy, while it has been shown that including structured information from the broader context of a social media post can further enhance the model's accuracy (Al-Sarem et al., 2019). Second, no study has attempted to understand why the model classifies a tweet as rumor or investigate what specific features drive the identification of a rumor. This is a missed opportunity since interpretable methods increase the transparency and reliability of a proposed technique and spur the uptake of your model by practitioners and decision makers (Coussement & Benoit, 2021). A potential reason for this absence of interpretations lies in the fact that rumor detection models are black boxes which do not easily provide insights into their predictions. This is even harder in the case of hybrid models using both textual and structured features. Whereas there are several model-specific and model-agnostic interpretation methods for the structured features (Bastos & Matos, 2021), unstructured textual features are often hard to interpret since they are transformed into embeddings or vector space models (Borchert at al., 2022). Especially in the case of deep learning methods, the use of advanced embeddings may increase predictive accuracy because they better capture the meaning of the text. However, this comes at the expense of interpretation

as they disable the direct link between vocabulary use and model output. This may explain why rumor detection model explanation has received no current academic interest. Third, comparatively little attention has been given to evaluating quality of the explanations of machine learning models. For example, Stevenson et al. (2021) assess the importance of unstructured textual features versus standard structured features for loan default prediction and also investigate the 20 most important words. However, they do not measure how reliable and stable their interpretations are across different samples.

To fill these gaps in literature, this paper proposes a methodology that tries to (1) accurately detect rumors, (2) explain what drives the identification of rumors and (3) evaluate the quality of the explanations. To detect rumors, we build several hybrid machine learning-based and deep learning-based models using both unstructured textual features and structured data. To explain the outcome of the hybrid models, we use the famous LIME method (Ribeiro et al., 2016) such that it allows for a model-agnostic explanation of the impact of individual words and structured features for both machine learning and deep learning approaches. Next, we propose several metrics to evaluate the quality of the resulting explanations. To increase the generalizability of our results we compare our rumor detection models and explanations across the well-known PHEME Twitter data set (Zubiaga et al., 2016) and two novel data sets regarding COVID-19 and the Amazon forest fires on Twitter. Overall, this study aims to answer the following research questions:

- *RQ1: Which hybrid machine learning or deep learning model performs best when including both unstructured and structured features for rumor detection?*
- *RQ2: How can we effectively measure the quality of the explanations and which type of model has the highest quality of the explanations?*
- *RQ3: Is there a trade-off between quality of explanations and model accuracy?*

By addressing these research questions, this study contributes to literature in several ways. First, we share two novel data sets about the COVID-19 pandemic and the Amazon forest fires on Twitter to the rumor detection community. Since there are only few data sets publicly available, of which the PHEME data set is the most important, we share our data sets with the research community such that other researchers can replicate our results and benchmark the performance of their implementations. Next, using two new data sets and the five PHEME data sets we compare a wide-range of machine learning and deep learning models using features related to the tweet text and structured features relating to the context of the tweet (e.g., length of the text, number of followers). Previous studies have solely focused on machine learning or deep learning methods or did not include structured features next to textual features, thus we are the first to compare both type of models including both features groups. Finally, we measure the quality of the explanations using fidelity and stability. Since there is currently no method to measure the stability of global feature importances across folds, we come up with a new metric to assess the stability of the explanations. Using these metrics, we investigate the trade-off between individual model performance and the quality of the explanations across a wide range of models.

Thereby, we provide insights to practitioners about not only the predictive performance of several machine learning and deep learning methods, but also their explanatory performance.

2. Related literature

This section provides an overview of the related work on rumor detection using social media data. Before discussing relevant literature, we would like to stress the difference between the commonly-used term ‘fake news’ and a ‘rumor'. The former refers to a piece of text that is created to be intentionally and knowingly false, while the latter is a piece of text that (still) has to be confirmed by official sources and is shared by social media users (Bondielli & Marcelloni, 2019). Hence, fake news are verifiable false messages deliberately created to go viral and spread through the social network (Zhang et al., 2019), whereas rumors are a particular form of information that are unverifiable at the moment of posting (Al-Sarem et al., 2019). In this work, we only focus on rumor detection and discard fake news detection (e.g., Paka et al., 2021; Yuan et al., 2021; Mridha et al., 2021). To sum up, we define a rumor as a piece of text that is shared throughout the social network that has a lack of verifiable evidence at the moment of posting (DiFonzo & Bordia, 2007).

Table 1 summarizes current rumor detection literature in social media. Accordingly, existing work can be categorized depending on the used data source(s), the included features, the type of modeling approach and whether or not they focus on model explainability. The PHEME data is arguably the most popular public data set in general rumor analysis literature (Al-Sarem et al., 2019), containing tweets about five events collected via the Twitter API (Zubiaga et al., 2016). Given the high inter-rater agreement, the PHEME data has been used by several studies (Ajoa et al., 2018) as a benchmark data set for comparing their proposed approaches. Other studies choose to gather novel data via the Twitter API to design rumor detection models about a specific event. For example, Al-Sarem et al. (2021) come up with a specific deep learning model to combat the spread of rumors in Arabic during the COVID-19 pandemic. The used features and modeling approach are largely intertwined in rumor detection models. If machine learning approaches are applied, several hand-crafted features are engineered. Textual features in that case refer to transformations such as bag-of-words that represent the unstructured message of the tweet into a structured format that can be inputted to a machine learning model (Qazvinian et al., 2011). Structured features can refer to both the content (e.g., POS tags or the number of hashtags) or the context of the tweet (e.g., characteristic of the poster like the number of tweets) (Kim et al., 2020). For deep learning approaches, the feature engineering process is performed by the model and only the message serves as input to the model (Kochika et al., 2018). In their study, Kumar et al. (2020) found that the best performing machine learning method was random forest and the best performing deep learning method a long short-term memory (LSTM). Recent work in OR not only focuses on good accuracy, but also on explainability to interpret the decisions made by the model. In this study, we want to investigate why the models classify a tweet as a rumor or not. Besides interpreting the model’s output, the quality of the explanations can also be assessed by looking at how trustworthy

the explanatory model approximates the black-box predictions (i.e., fidelity) and the stability of the explanations across different samples (Ramon et al., 2021).

Table 1: Overview of rumor detection literature in social media

	Data		Features		Models		Explainability	
	PHEME	New data	Text	Structured	Machine learning	Deep learning	Interpret	Quality
Qazvinian et al. (2011)		X	X	X	X			
Ajoa et al. (2018)	X		X			X		
Kochika et al. (2018)	X		X			X		
Hamidian & Diab (2019)[1]			X	X	X			
Ke et al. (2020)		X	X			X		
Kim et al. (2020)	X		X	X	X			
Kumar et al. (2020)[2]	X		X	X	X	X		
Alqurashi et al. (2021)		X	X		X	X		
Al-Sarem et al. (2021)		X	X			X		
Almars et al. (2022)		X	X			X		
Our study	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**

[1]This study uses the data set introduced by Qazvinian et al. (2011).
[2] In this study the deep learning models only use the textual features as input. The combination of structured and textual features only applies to the machine learning models.

Table 1 shows that no study has built machine learning and deep learning-based rumor detection models with both textual and structured variables and compared their performance across the PHEME data set and novel data sets. Whereas the study of Kumar et al. (2020) already performed a benchmark for the PHEME data sets, their study did not include any novel data sets nor did it use a combination of both features for the deep learning models. Moreover, attention-based models (Vaswani et al., 2017) were

not included in their work and recent studies (Ke et al., 2022; Almars et al., 2022) demonstrate that sequential learners like LSTMs are often outperformed by these models. Several recent OR applications show that the combination of unstructured and structured features can also increase the predictive accuracy of deep learning models considerably (Borchert et al., 2022; Stevenson et al., 2021). Hence, there is no proper way of knowing what approach performs best on the benchmark data set or the novel data sets when including both textual and structured features. Therefore, this study shares two novel data sets regarding rumor detection to the research community: one about the COVID-19 pandemic and one about the Amazon Forest fires. Using these two new data sets and the five PHEME data sets, the first aim of this study is to set-up a benchmark study that compares machine learning, deep learning and transformer models with both feature types (textual and structured) as inputs.

Table 1 also clearly shows that no single study has tried to explain the outcome of a rumor detection model or investigated what drives a model to flag a message as a rumor. In general, explainability is the degree to which a human can interpret the decisions made by the model (Miller, 2017). In recent years, the interest in model explainability has increased considerably in the OR domain (Bücker et al., 2022). This is mainly driven by the omnipresence of black-box models in our daily lives and business operations, which makes it necessary for practitioners and decision makers to understand and validate what the model has learned. A highly explainable model also reduces model uncertainty and increases the probability of adoption within the business (Coussement & Benoit, 2021). Explainable techniques can be categorized according to (1) how detailed the aspects of model should be explained (i.e., only detect the most important variables or also assess their relationship with the response?), and (2) whether the explanations should investigate model behavior on the global or the local level (Bücker et al., 2022). Most studies in OR focus on how the variables contribute and affect the model predictions on the global level. For example, Borchert et al. (2022) assess permutation-based feature importances to find out what textual content has the most impact on business failure prediction. Other authors take it one step further and do not only investigate the global impact of textual features, but also the local impact of individual words. For example, in the field of loan default prediction, Stevenson et al. (2021) apply LIME to the 250 instances where the unstructured data added most predictive value. The great advantage of LIME for textual data is that it allows to interpret both the impact size and direction of individual words on the instance prediction for any black-box model. Since it is utterly important for policy and decision makers to know which individual words are associated with rumors, and whether this is positive or negative, LIME is a good candidate as a model interpretation tool for rumor detection. Moreover, a proper interpretation method for rumor detection should not only explain individual cases but also provide a global understanding of the model's behavior, making LIME an even more adequate candidate as aggregating the local model's coefficients across individual cases allows for global model explainability. However, what makes LIME truly an ideal candidate is the necessity to tackle both textual and structured features. Many other explanation methods that offer both local and global explanations leverage the features as inputted into the model. For deep learning architectures, the text is

often transformed into embeddings which serve as features and are not humanly interpretable. Methods that try to make these text representations interpretable are often model-specific (e.g., attention heatmaps). LIME, on the other hand, allows for human-interpretable representations in a white-box model independent from the feature representations used in the complex black-box model. Therefore, our explanatory model uses a LIME implementation that provides global model-agnostic explanations of both textual and structured features and is easily interpretable.

Finally, studies regarding explainability in OR often only focus on comprehensibility. This means that humans should be able to easily understand the explanations. However, when designing an explainable tool other issues should be considered, such as accuracy, fidelity, stability and novelty, among others (Molnar, 2022). From those properties, the fidelity and the stability of the explanations are arguably the most important since they define the overall quality of the explanations (Ramon et al., 2021). Fidelity refers to how well the explanations approximate the black box model (Martens et al., 2007), whereas stability refers to the change in explanations when repeated over different training samples (Visani et al., 2020). Especially in the case of LIME where a local surrogate white box model is built to approximate the original black box model, the fidelity and the stability of explanations constitute a well-known issue that should be taken into consideration (Visani et al., 2020). Authors who have attempted to measure the stability of the explanations often look at local robustness of their methods. For example, Alvarez-Melis et al. (2018) compute the local robustness of LIME by randomly sampling 200 instances from the test set. However, analysts and decision makers are particularly interested in the global behavior of the model in the model selection and model evaluation (Bücker et al., 2022). When evaluating the most appropriate interpretable model, the evaluation should thus take place at the global and not the local level. Since there is no unified method to measure the global stability of the explanations, the third aim of this study is to propose a novel metric to measure overall stability of the variable rankings produced across different folds. The final aim of this study is to assess the quality of the explanations (fidelity and stability) and benchmark them across different data sets and models, and to see whether there is a trade-off between model accuracy and the quality of the explanations.

3. Methodology

Figure 1 outlines the process of our proposed methodology. First, the two new data sets (indicated in purple) are collected via the Twitter API and annotated by two independent labelers. The five PHEME data sets (indicated in blue) are publicly available and can be used as such. Next, the (unstructured) textual features, and content- and context-based structured features are constructed for all data sets. For the textual features, different representations are necessary depending on the modeling approach. For example, transformers only need the raw text as input whereas machine learning models require document vectors. In a next step, a benchmark study is performed across all seven data sets using

several machine learning and deep learning models. To ensure that the results are robust, we follow previous literature and employ five times two-fold cross-validation (5x2fcv; Dietterich, 1998), which consists of randomly splitting the data into two distinct non-overlapping samples of equal size (i.e., 50/50 distribution). The first sample is used to train the model (i.e., training set), whereas the second sample is used to evaluate its performance (i.e., test set), and vice versa. If the classification model necessitates hyperparameter tuning, the training set is again equally split into a training and a validation set. In the case of machine learning models, a random search is used to find the optimal parameters. For tuning the layers of the deep learning models, we use a Bayesian optimization process. To evaluate the predictive performance of our rumor detection models and to optimize our hyperparameter settings, we compute the AUC (Area Under ROC Curve), as it is commonly adopted in prior rumor detection studies (Ajao et al., 2018; Kumar et al., 2020; Zubiaga et al., 2017). Following the hyperparameter tuning, the models are retrained on the full training set with the optimal parameter settings. By repeating this process 5 times in the 5x2fcv scheme, we obtain 10 performance values. The reported results are the median AUC values over the 10 cross-validation runs. Finally, besides predictive performance, we also evaluate explanation quality using the proposed fidelity and stability metrics. The rumor detection models are interpreted using LIME and the quality of the explanations is evaluated across all folds and data sets. A detailed overview of all the steps in our proposed methodology is provided in the next sections. The implementation of our methodology is entirely performed in Python and is made publicly available via https://github.com/bram-janssens/ExplanationQuality.

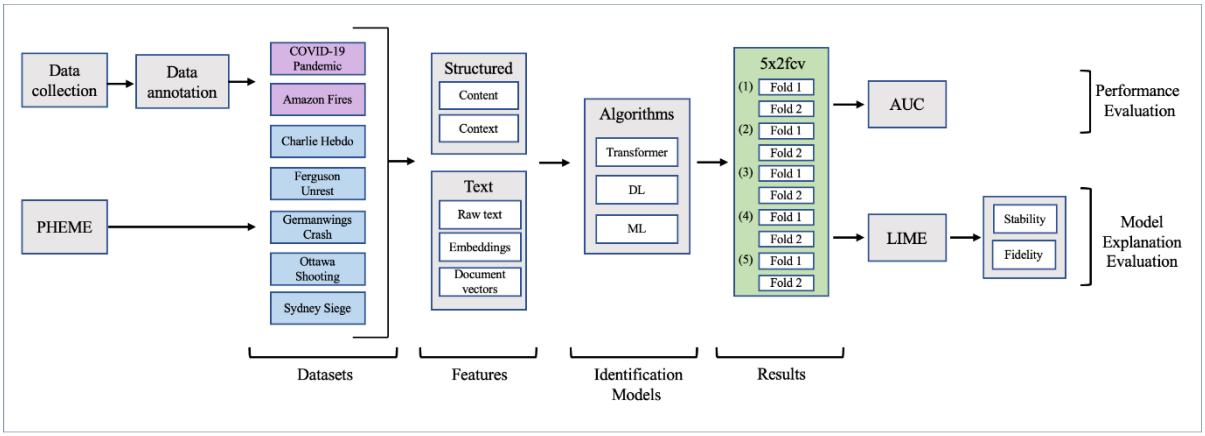

Figure 1: Proposed research methodology

3.1. Data

Data collection

The first five data sets originate from the PHEME data set, developed by Zubiaga et al. (2016). Specifically, we use the data sets related to the following events: (1) the Charlie Hebdo shooting, which took place on January 15th, 2015, in Paris, (2) the Ferguson unrest, which resulted from the deathly shooting of Michael Brown by a police officer in August, 2014, in Ferguson, Missouri, (3) the

Germanwings plane crash in the French Alps on March 24th, 2015, (4) the Ottawa shooting that took place on October 22nd, 2014, on Parliament Hill in Ottawa, and, finally, (5) the Sydney siege, during which 18 people were held hostage in Sydney, on December 15th, 2014 (Zubiaga et al., 2016). These data sets are all related to breaking newsworthy topics, during which new rumors were expected to emerge and were not known a priori.

Besides the five aforementioned PHEME data sets this study introduces two new rumor data sets. The first new data set is related to the Amazon rainforest fires that took place in 2019, whereas the second data set is about the COVID-19 pandemic. To collect a wide range of tweets related to these topics, tweets were collected via the Twitter API based on a number of related hashtags. With regard to the Amazon rainforest fires, tweets were gathered using #PrayforAmazonas, #AmazonRainforest, and #AmazonFire on a daily basis between August 21, 2019 and September 27, 2019. For the COVID-19 data sets, tweets were gathered based on #CoronaOutbreak, #CoronaVirus, #CoronaVirusOutbreak, #COVID19, #COVID-19, #COVID2019, and #SARSCoV2, between February 12, 2020 and June 15, 2020. The collected data consists of the message of the tweet itself (i.e., the raw text) as well as contextual information (e.g., number of characters, language, user, date of the tweet, etc.) that can be stored as structured information. Considering the large volume of tweets and in line with the PHEME data sets, we follow the recommendation of Zubiaga et al. (2016) to sample our new data sets by only retaining those tweets with a sufficient number of retweets, where the cut-off is based upon the overall popularity of the topic (i.e., 75 for the COVID-19 data set, 1 for the Amazon fires data set). The number of retweets is especially relevant as it is indicative of the traction a rumor is gaining with the general public (Zubiaga et al., 2016). This results in a final data set of 1,392 unique English tweets on the Amazon rainforest fires and a data set of 4,612 unique English tweets about the COVID-19 pandemic.

Data annotation

Whereas the five PHEME data sets already contain the necessary labels, the tweets in the Amazon fires and COVID-19 data sets still needed to be annotated as being a rumor or not. We follow the seminal work of Zubiaga et al. (2016) and define a rumor as a non-private assertion that at the time of tweeting cannot be verified because it has not been confirmed by official sources. This definition contains two important elements, namely a rumor should be (1) an *assertion* that is (2) *unverified* at the time of posting. An assertion (Searle, 1975), as opposed to expressions or recommendations, is a statement in which the writer commits to the truth of the message. For example, in the context of the COVID-19 pandemic 'We are going into lockdown this Friday' is an assertion, while 'I am scared we might be going into lockdown this Friday' is an expression. We qualify an assertion as verified if, *at the time of tweeting*, it has been published by Reuters or BBC News. Following these two important guidelines, each tweet was carefully annotated by two independent experts (Meire et al., 2019) in order to determine the tweet label (i.e., rumor or not). The entire and detailed annotation guidelines can be found in Appendix A of the supplementary materials. The inter-rater reliability between the annotators

is determined by means of Cohen's kappa measure (Meire et al., 2019). After the first round of labeling, Cohen's kappa equaled 0.9034, indicating a high level of agreement between the two annotators. To determine the final value of the label (1 = rumor, 0 = no rumor), the experts reiterated over the tweets they disagreed on until a consensus was reached. Appendix B presents some examples of tweets labeled as rumors and non-rumors. The original data for this study are available at Mendeley Data (COVID19 data set: http://dx.doi.org/10.17632/xcz8vb448y.1; Amazon fires data set: http://dx.doi.org/10.17632/m7k4gsffry.1).

Table 2 displays the (relative) presence of rumors across the data sets included in this study. For the five PHEME data sets (i.e., Charlie Hebdo, Ferguson Unrest, Germanwings Crash, Ottawa Shooting, and Sydney Siege), this variable is available in the data set developed by Zubiaga et al. (2016). We notice that not only the total size of these data sets is different, but also the relative number of rumors across the seven data sets (i.e., ranging from 11% to 53%). Generally, the percentage of rumors in the PHEME data sets appears to be higher than in the newly introduced data sets.

Table 2: Distribution of rumors across the seven data sets.

Data set	Rumors	Non-rumors	Total
PHEME			
Charlie Hebdo	458 (22%)	1,621 (78%)	2,079
Ferguson Unrest	284 (25%)	859 (75%)	1,143
Germanwings Crash	238 (51%)	231 (49%)	469
Ottawa Shooting	470 (53%)	420 (47%)	890
Sydney Siege	522 (43%)	699 (57%)	1,221
New			
COVID-19 Pandemic	485 (11%)	4,127 (89%)	4,612
Amazon Fires	184 (13%)	1,208 (87%)	1,392

3.2. Variables

The rumor detection models include two types of features: unstructured textual information (i.e., message of a tweet) and structured features. The latter consist of content- and context-related features, and have proven to enhance model performance in multiple rumor analysis tasks (Qazvinian et al., 2011; Zubiaga et al., 2017). Content-related features are linked to the content of a tweet. For example, they describe whether certain words or punctuation marks are used in a tweet. The content-related variables, as well as an explanation of their potential value, are presented in Table C.1 in the supplementary materials. Context-related features are variables characterizing (the activity of) a Twitter account and can, therefore, add information with regard to source reliability (Zubiaga et al., 2017). The context-related features used in this study, along with the rationale behind them, are presented in Table C.2 in the supplementary materials.

With regard to the textual features, different classification models each require different levels of input data. The input of the transformers simply consists of the raw text of the tweets. With regard to other deep learning models, the unstructured textual data has to be transformed into a feature vector representation by means of word embeddings. Word embeddings represent the words of a corpus in a multidimensional vector space such that words with similar meanings have similar vector representations and are able to grasp syntactic and semantic similarities between words (Le & Mikolov, 2014). To accelerate the training process of deep learning models and cope with the limited sample sizes of the data sets (Craja et al., 2020), we use a pre-trained set of word embeddings. Specifically, we use Stanford's GloVe algorithm trained on Twitter data (Pennington et al., 2014) with the number of dimensions set to 200. The 200D-representation is the highest feasible dimensionality, thus providing the most nuanced differences of underlying word meanings. Finally, machine learning models are not capable of handling sequential data such as individual words and thus the word embeddings should be aggregated on document level. Document vectors are defined as the sum of the individual word vectors resulting from the application of the pre-trained word embeddings (Le & Mikolov, 2014).

3.3. Rumor detection models

The considered ML models have proven to yield superior performance in several rumor detection tasks (Bondielli & Marcelloni, 2019), as well as on social media data (Ballings et al., 2015). In terms of single classifiers, we consider logistic regression with lasso regularization (LR) and support vector machines (SVM). LR has been used in a number of rumor analysis studies, particularly for rumor stance classification (Bondielli & Marcelloni, 2019). Furthermore, it has also been employed as a baseline model in rumor identification tasks (Qazvinian et al., 2011). Whereas LR uses a logistic function to model the dependent variable, SVM separates the data into classes by finding a separating hyperplane. It is capable of learning complex non-linear functions by using the kernel trick. SVM is one of the most popular methods for rumor veracity classification (e.g., Liu et al., 2015). Besides these single classifiers, we also include an ensemble method, namely random forest (RF). In current literature, RF is used for a variety of tasks within rumor analysis (e.g., Liu et al., 2015). Kumar et al. (2020) even identified RF as the top performer amongst several ML and simple DL models for rumor identification. A summary of the hyperparameter settings that are tuned using a random search of these models can be found in Appendix D of the supplementary materials.

For the DL models, we include models that can deal with sequential data, as we want to maintain the importance of the word order in our analysis. Hence, we consider convolutional neural network (CNN), long short-term memory (LSTM), and gated recurrent unit (GRU). CNNs are neural networks that consist of an input layer, an output layer, and a series of hidden layers. The latter transform the data by means of pooling and convolution operations (Bondielli & Marcelloni, 2019). CNNs have proven to be suited for distinct rumor analysis tasks (e.g., Chen et al., 2017). LSTM and GRU are both special

types of recurrent neural networks (RNNs) that solve the vanishing gradient problem (Zinovyeva et al., 2020). They consist of various gates and hidden states to control the flow of information that is passed along the model. LSTM consists of a forget gate, an input gate and an output gate (Fisher & Krauss, 2018); GRUs are less complex and only have an update and reset gate (Kraus et al., 2020). These gates enable sequential dependencies to be learned by the model. There are strong indications in current literature that LSTM and GRU are highly performant for rumor identification (Kumar et al., 2020). Since we combine the textual information (i.e., represented by the word embeddings) and the structured data (i.e., context- and content-related features) in our model, we consider specific hybrid deep learning architectures. Hybrid deep learning models have a multichannel architecture that consists of two parallel neural networks, as presented in Figures 2 and 3. The first channel is a multilayer perceptron (MLP), consisting of a dense layer, that takes the metadata as its input. This dense layer is added to capture possible non-linear and interactions effects between the structured input variables and the dependent variable. The number of units of the MLP layer is tuned during a Bayesian optimization process. The second channel takes the word sequences as an input, and consists of an embedding layer (i.e., initialized with pre-trained GloVe embeddings) followed by a convolutional, or LSTM/GRU block. A convolutional block consists of a one-dimensional convolutional layer with a ReLu activation function, a max pooling layer, and a flatten layer to reduce the dimensionality of the output of the second channel. An LSTM (GRU) block consists of an LSTM (GRU) layer, followed by a dropout layer to reduce overfitting and improve the generalization of the neural network. Similar to the convolutional block, a flatten layer is added. As can be seen in Figure 2, the outputs of the two channels are concatenated and serve as the input of the dense classification layer, after which a sigmoid activation function is applied. An early stopping monitor and regularization are applied to all deep learning models to avoid overfitting. A summary of the hyperparameters that are tuned during Bayesian optimization of the DL models can be found in Appendix D of the supplementary materials.

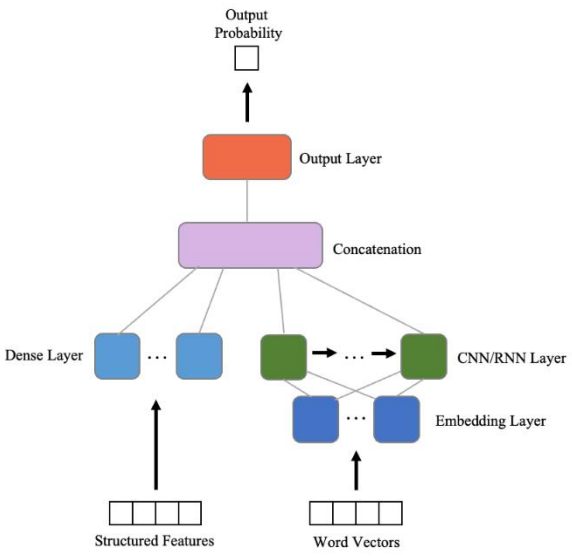

Figure 2: Schematic Overview RNN & CNN architectures

As a result of their superior performance in numerous tasks, transformers have gained a lot of popularity in the field of NLP. They can be interpreted as an implementation of the encoder-decoder architecture that transforms a sequence of words into a new sequence while using the attention mechanism (Vaswani et al., 2017). The latter is capable of learning which surrounding words are important with regard to the prediction task at hand. This enables deep transformers to 'understand' context much better than previous NLP algorithms, resulting in high performances. However, learning complicated tasks requires a large amount of data, which is often not available. Luckily, there are several pre-trained transformers in which the weights fitted in the encoder part used on large data sets can be transferred to other tasks (i.e., the 'encoding' of language). This type of transfer learning makes pre-trained transformers highly suitable for complex text classification tasks with limited training samples such as rumor detection. In this study we consider a BERT model (Devlin et al., 2018), which has recently gained a lot of interest in the OR literature (e.g., Borchert et al., 2022). BERT uses the encoder to learn the latent representation of the input text. It does so by learning two specific tasks: Masked Language Model (MLM) (i.e., learning which words are masked or missing) and Next Sentence Prediction (NSP) (i.e., predicting, based on the input sentence, whether a suggested sentence is the actual true sentence). These tasks are revolutionary as they allow to learn context of text bi-directionally (i.e., both words coming after and before the masked words are used) as well as across sentences. The exceptionally large amounts of data needed by transformer models are often not available in OR research, and are clearly lacking in data sets on rumor identification (cf. Table 2). This leaves two types of methodologies: fine-tuning all the parameters of the architecture or using frozen pre-trained weights in the architecture (Peters, Ruder & Smith, 2019). Interestingly, high variability is observed when fine-tuning transformer architectures on limited sample sizes (Mosbach et al., 2020; Zhang et al., 2020), while using frozen pre-trained weights is shown to perform highly competitive against these unstable fine-tuned architectures (Peters, Ruder & Smith, 2019). Accordingly, we also adopt such a frozen pre-trained weights methodology, given the limited sample sizes available in rumor detection literature. Similar to other deep learning models, a hybrid architecture that consists of two parallel channels is used, as depicted in Figure 3. The first channel is again an MLP, consisting of a dense layer, that takes the structured features as its input. The BERT model in the second channel receives the raw text of the tweets as an input and outputs a BERT embeddings layer, which uses the pre-trained weights as discussed before. Note that this approach is different than using pre-trained BERT embeddings as features in a downstream classifier, in which the features do not change based on the specific context of tweets. In our approach we use pre-trained weights and pass the raw text through the full BERT architecture enabling to take into account this context. The main distinction compared to the other DL models lays in the existence of an additional hidden layer between the text representations and the concatenated layer. The reason that our BERT model has this additional layer, is the fact that the weights of the BERT embedding block are frozen (as opposed to the other DL architectures that do not use any prior knowledge). This hidden layer allows to leverage the high predictive performance of the frozen

weights of the pre-trained model, while still allowing our classifier to learn new relationships relevant for the task at hand (i.e., rumor detection). We use the 'base' BERT implementation trained on lower-cased English text, with a 12-layer, 768-hidden, 12-heads architecture, the ADAM optimizer and the learning rate set to 0.00004 (Devlin at al., 2018). The outputs of the two channels are concatenated and serve as the input of a dense classification layer. A Bayesian optimization scheme similar to the other DL architectures is followed, with the hyperparameters also listed in Appendix D of the supplementary materials.

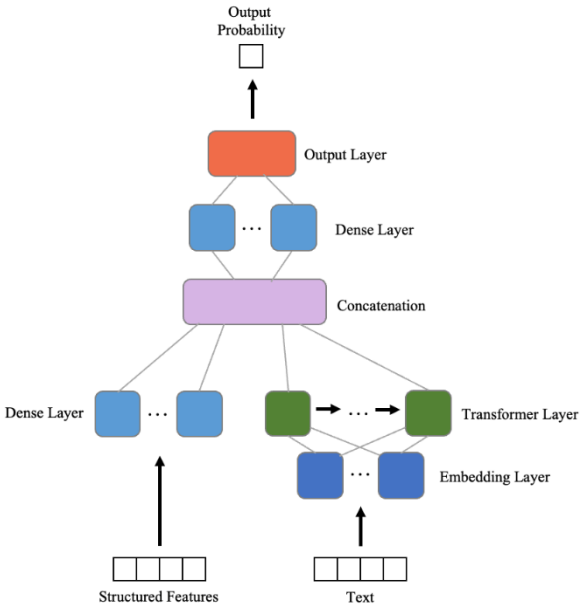

Figure 3: Schematic Overview Transformer architectures

3.4. Model evaluation

To validate predictive performance, the AUC measure is reported. However, reliable and stable explanations are also indispensable to a user-oriented and robust decision support system on rumor detection. Accordingly, Figure 4 shows the pseudocode of our proposed methodology to evaluate the quality of the explanations of individual classifiers with highly varying architectures across folds. The pseudocode consists of two major parts: (1) the Local Interpretable Model-agnostic Explanations (LIME; Ribeiro et al., 2016) framework and fidelity (line 1 to 17), and (2) the stability of the explanations (line 18 to 30).

Explainability Evaluation Algorithm

1: **Set** $fidelities$ **To** empty list
2: **Set** $RankingFold$ **To** empty list
3: **For** $i = 1$ to k folds **do**:
4: **Get** the training data D_{fold} and the trained black-box model f from fold i
5: **Set** $FeatureImportances$ **To** empty list
6: **For** $observation$ **in** D_{fold} **do**:
7: Create 1000 permutations $\boldsymbol{x_{LIME}}$ from $observation$
8: Create 1000 interpretable representations $\boldsymbol{x_{INTERPRETABLE}}$ from $\boldsymbol{x_{LIME}}$

9: **Calculate** $similarities = \boldsymbol{x_{LIME}}^{T}.observation$
10: **Predict** $\boldsymbol{y_{LIME}} = f(\boldsymbol{x_{LIME}})$
11: **Get** the m (20) most important features by fitting a ridge regression between $\boldsymbol{y_{LIME}}$ and $\boldsymbol{x_{INTERPRETABLE}}$ with $similarities$ as weights
12: **Fit** ridge regression $\boldsymbol{y_{LIME}} = g(\boldsymbol{x_{INTERPRETABLE}})$ with m as considered features
13: **Calculate** $fidelity = 1\text{-}\frac{Residual\ Sum\ of\ Suares(g)}{Total\ Sum\ of\ Squares(g)}$
14: **Add** $fidelity$ **To** $fidelities$
15: **Calculate** FI which are the standardized coefficients from g
16: **Add** FI **To** $FeatureImportances$
17: End For;

18: **Calculate** $AggregatedFI =$ Sum all absolute values from $FeatureImportances$
19: **Rank** $AggregatedFI$
20: **Get** $top10 =$ highest 10 importances from $AggregatedFI$
21: **Add** $top10$ **To** $RankingFold$
22: End For;

23: **Set** $PairwiseJaccard$ **To** empty list
24: **For** $i = 1$ to k folds **do**:
25: **For** $j = 1$ to k folds **do**:
26: **If** $i < j$:
27: **Calculate** $JS = J(RankingFold_i, RankingFold_j)$
28: **Add** JS **To** $PairwiseJaccard$
29: End For;
30: End For;

31: **Calculate** $OverallFidelity =$ average$(fidelities)$
32: **Calculate** $OverallStability =$ average$(PairwiseJaccard)$

Figure 4: Pseudocode of the evaluation of model explainability

Interpreting the combination of structured and textual features is more complex compared to 'traditional' data as these inputs are typically structured in a numeric format and are relatively low-dimensional (Ramon et al., 2021). Moreover, the direct interpretation of the hidden features in deep learning architectures is hard, as the interpretation of a specific embedding is quasi nonsensical in human language. These issues call for a specific model-agnostic interpretation framework that performs well for high-dimensional textual inputs, which made us opt for the LIME framework. The main reasons are: (1) the model-agnostic nature of the method enabling the comparison of algorithms that use very different data representations, and (2) its relative computational efficiency (Stevenson et al., 2021) compared to alternative model-agnostic methods (e.g., SHAP; Lundberg & Lee, 2017). This computational efficiency is especially important given the inclusion of complex deep learning architectures across seven different data sets and ten folds. The great fit of LIME towards high-dimensional textual data and computational efficiency has resulted in the method being very popular among studies which focus on interpretability in decision support models (e.g., Zinovyeva et al., 2020; Stevenson et al., 2021).

The main idea behind LIME is that a complex model may not be approximated by simple models globally, but that simple models can reliably mimic the complex model locally. Therefore, local permutations are created around each individual original observation (lines 7 and 8 in Figure 4), which are then used as the input to create new point predictions by the black box model f (line 10). These new synthetic observations, which all exist in the local proximity of the original $observation$, are then used to fit the interpretable simple model g (line 12). When implementing LIME with unstructured and structured features there is a difference between $\boldsymbol{x_{LIME}}$ and $\boldsymbol{x_{INTERPRETABLE}}$, used for the complex and simple models, respectively. $\boldsymbol{x_{LIME}}$ is a model-specific representation (e.g., GloVe or BERT embeddings) as discussed in Section 3.2, while $\boldsymbol{x_{INTERPRETABLE}}$ is a model-agnostic representation which in our case depicts the textual information in a term frequency vector and the structured features in their original data representation, which are easy to interpret. Thus, $\boldsymbol{x_{LIME}}$ is the permutation representation as interpreted by the black box model, while $\boldsymbol{x_{INTERPRETABLE}}$ is the interpretable permutation representation which is used in the LIME framework for the white box model. To understand this distinction, consider the following example: 'Big State is fueling the COVID hoax'. Each algorithm will create a representation based on its architecture (i.e., GloVe embedding or BERT embedding) which is multi-dimensional representation of this sentence (e.g., fictive GloVe vector of 200 dimensions: [0.01551, -0.11561, …, 2.01515]). However, these representations are not humanly interpretable, and neither are the corresponding permutations around these values (i.e., $\boldsymbol{x_{LIME}}$) which are used to predict $\boldsymbol{y_{LIME}}$. Therefore, when building the interpretable model, these permutations are translated to humanly interpretable representations $\boldsymbol{x_{INTERPRETABLE}}$ as depicted in Table 3. In our case permutation 1 would get the highest weight in our interpretable LASSO regression as it is the most similar permutation to the initial observation (without considering context features). For a more elaborate discussion on LIME, we refer the reader to the original work by Ribeiro et al. (2016).

Table 3: Visualization $\boldsymbol{x_{INTERPRETABLE}}$ text representations.

Permutation	big	state	is	fueling	the	covid	hoax
1	1	1	1	1	0	1	1
2	0	0	1	0	1	0	1
3	1	0	1	0	0	0	1

While explanation methods such as LIME are extremely useful, researchers should not blindly follow the explanations, since explanation methods can be misrepresenting actual model behavior. Rather, the main goal of a surrogate model g should be to mimic the behavior of the complex model f as good as possible (Carvalho et al.,2019). In literature, this is known as fidelity or trustworthiness (Martens et al., 2008), which can be measured by the extent to which the simple model g is capable of capturing the complexities of the global model f. Thus, the goodness-of-fit of the model g between the interpretable data representation $\boldsymbol{x_{INTERPRETABLE}}$ and the outputs $\boldsymbol{y_{LIME}}$ as produced by the complex

model f is an excellent way of capturing this information. Accordingly, we will measure the coefficient of determination (R^2) of each ridge regression model g across all individual explanations (line 13) (Shankaranarayana & Runje, 2019). These individual fidelity scores are then averaged across all individual local explanations using the following formula:

$$Fidelity = \frac{1}{K}\sum_{k=1}^{K}\frac{\sum_{i=1}^{N_k} R_i^2}{N_k} = \frac{1}{K}\sum_{k=1}^{K}\frac{\sum_{i=1}^{N_k}(1-\frac{\sum_{j}^{P=1000}(y_{LIME_i,j}-\hat{y}_{g_i,j})^2}{\sum_{j}^{P=1000}(y_{LIME_i,j}-\bar{y}_{g_i})^2})}{N_k},$$

with N_k the number of observations in fold K and R_i^2 the coefficients of determination of observation i in fold K. Note how this consistently compares the $y_{LIME_i,j}$ predictions of the complex model f on permutation j of observation i with the predictions $\hat{y}_{g_i,j}$ from the model g_i on permutation j of observation i, with g_i being the simple model which is fitted on these $\boldsymbol{y_{LIME_I}}$ values.

Besides being trustworthy, explanations should also remain stable across different training rounds, as unstable decision making can induce large unnecessary costs (Van Belle et al., 2022). Stability is a term that is frequently used in the OR community, but with different meanings. Therefore, we detail what stability means in the current context and distinguish it from previous work. First, we focus on explanation stability rather than prediction stability. While the latter is well-established within the research community (e.g., Van Belle et al., (2022)) and focuses on getting similar predictions for instances when slightly perturbing the instance or similar predictions based on different training sets, the former focuses on getting similar interpretations. Second, there is a difference between local or global stability, just like local or global explanations. LIME, and also SHAP, have repeatedly been shown to be unstable when generating instance (or local) interpretations (Visani et al., 2022; Alvarez-Melis & Jaakkola, 2018). That implies that running LIME multiple times for the same instance can yield different interpretations. Previous work in computer vision models has already investigated the stability of local explanations (Alvarez-Melis & Jaakkola, 2018). On the other hand, global stability for machine learning and deep learning models has received only scant attention by the research community, and has mainly been applied to rule-based explanations (Ramon et al., 2021). However, many OR and business analytics studies give primary attention to global explanations (Bastos & Matos, 2022).

Following the aforementioned discussion, we define explanation instability as the phenomenon where slight changes in the training data result into different global explanations. The possible existence of such instability is especially worrisome in studies who follow a cross-validated experimental set-up, where algorithmic interpretation is typically based upon one fold (Janssens et al., 2022). However, no prior study investigated whether global model explanations are stable across folds and whether different algorithms tend to deliver more stable interpretations compared to others. Accordingly, we investigate the stability of global model explanations by looking at how these explanations deviate across folds. More specifically, we look at the stability of the feature importance rankings, as the identification of the top N drivers is inherent to how the LIME framework is structured in order to facilitate human

interpretation. Note that this approach can also be used with other importance rankings, such as the permutation-based feature importances generated from tree-based ensembles.

To define stability, we can (1) evaluate and compare sets of variables which are deemed important or (2) compare the size of specific importance metrics for all variables (see Visani et al. (2021) for a similar distinction for local stability). We focus on the former, as the size of importance metrics may easily differ and make it more difficult to compare across algorithms. Since we want to compare sets of important variables, we can rely on the well-known Jaccard coefficient. In essence, this metric calculates the similarity between two sets of explanations. Following research regarding the stability of rule-based explanations (Ramon et al., 2021), we calculate the pairwise Jaccard coefficients between all individual folds and average these. Formally, the equation becomes:

$$Stability_N = \frac{\sum_{f_1<f_2}^{K} J(F_{f_1}, F_{f_2})}{\binom{K}{2}} = \frac{\sum_{f_1<f_2}^{K} \frac{\left|F_{f_1} \cap F_{f_2}\right|}{\left|F_{f_1} \cup F_{f_2}\right|}}{\frac{K!}{2!\,(K-2)!}},$$

with K the number of folds, N the top number of variables considered for calculating stability, and $F_{f_1} \cap F_{f_2}$ the intersection of variables which occur in the top-N variable importances of both folds f_1 and f_2, while $F_{f_1} \cup F_{f_2}$ is the union of these feature sets. For $K = 2$, this stability metric is equal to the Jaccard index, while it takes the average of all pairwise Jaccard indices when $K > 2$. A practical example of the use of this stability measure is shown in Appendix E of the supplementary materials. In our approach, the pairwise Jaccard scores are calculated (line 23-30 in pseudo-code) from the aggregated feature importances per fold (lines 18-22), which are the summed absolute coefficient values of each individual simple model g_i. We set N equal to 10, as a limited set of features is also often reported in global explanations (Bastos & Matos, 2022). Finally, both fidelity and stability are reported for each algorithm-data set combination (lines 31 to 32).

4. Results

4.1. Predictive performance

Table 4: Predictive performance (median AUC) of each algorithm across all seven benchmarked data sets. Minimum and maximum values across folds indicated between brackets.

	Charlie Hebdo	Ferguson Unrest	Germanwings Crash	Ottawa Shooting	Sydney Siege	COVID Pandemic	Amazon Fires	**Average Rank**
LR	0.8864	0.8190	0.8785	0.8873	0.8437	0.7915	0.7373	3.71
	(0.8606)	(0.7833)	(0.8471)	(0.8670)	(0.8100)	(0.7596)	(0.7053)	
	(0.9077)	(0.8734)	(0.9141)	(0.9001)	(0.8706)	(0.8066)	(0.7621)	
SVM	0.9041	0.8604	**0.8950**	**0.9367**	**0.8736**	0.7715	0.7889	**2.00**
	(0.8691)	(0.7902)	(0.8150)	(0.9025)	(0.8267)	(0.6867)	(0.7518)	
	(0.9284)	(0.8930)	(0.9149)	(0.9493)	(0.9010)	(0.8115)	(0.8154)	

RF	0.8960	0.8526	0.8917	0.9253	0.8814	0.7641	**0.7948**	2.29
	(0.8860)	(0.8288)	(0.8691)	(0.9088)	(0.8620)	(0.7501)	(0.7724)	
	(0.9094)	(0.8765)	(0.9320)	(0.9337)	(0.9017)	(0.7902)	(0.8106)	
H-CNN	0.8405	0.7120	0.6714	0.7467	0.7538	0.7431	0.7320	6.71
	(0.7248)	(0.4801)	(0.5356)	(0.6827)	(0.6648)	(0.7095)	(0.6869)	
	(0.8585)	(0.7776)	(0.7880)	(0.8208)	(0.8196)	(0.7770)	(0.7803)	
H-LSTM	0.8733	0.7584	0.8031	0.8987	0.8212	0.7535	0.7284	5.43
	(0.8189)	(0.7348)	(0.6704)	(0.8822)	(0.8039)	(0.7154)	(0.6784)	
	(0.8988)	(0.8091)	(0.8845)	(0.9161)	(0.8567)	(0.7645)	(0.7688)	
H-GRU	0.8801	0.7522	0.7270	0.8665	0.8074	0.7545	0.7303	5.71
	(0.8317)	(0.7185)	(0.6160)	(0.7078)	(0.7994)	(0.7339)	(0.6801)	
	(0.8910)	(0.7966)	(0.8400)	(0.9063)	(0.8420)	(0.8019)	(0.7720)	
H-BERT	**0.9100**	**0.8901**	0.8676	0.9200	0.8684	**0.8279**	0.7913	2.14
	(0.8944)	(0.8611)	(0.8007)	(0.8854)	(0.8441)	(0.5081)	(0.7419)	
	(0.9177)	(0.9177)	(0.8933)	(0.9359)	(0.8946)	(0.8408)	(0.8347)	
# *obs*	2,079	1,143	493	890	1,221	4,612	1,392	

Table 4 depicts the median AUC scores of the considered algorithms across the seven benchmarked data sets. The lowest and highest scores across the ten folds are reported between brackets. The top performing algorithm per data set is indicated in bold and underlined, the average rank of the algorithms across the data sets is reported in the last column. With regard to the PHEME data sets, we observe predictive performances in line with previous studies using hybrid models, which report F1 scores ranging between 0.84 and 0.96 using traditional learners (Kim et al., 2022), and accuracies up to 0.91 using deep learning architectures (Kumar et al., 2020). The weak performance of typical deep learning methods is remarkable, with convolutional neural networks, long short-term memory networks, and gated recurrent unit networks consistently being the weakest performers across all seven data sets. This is especially surprising given the popularity these methods have received during recent studies on rumor detection (see Section 2. Related literature). However, most of these studies *only* considered deep learning architectures, and did not compare their performance with more traditional machine learning approaches. Only one study (Kumar et al., 2020) compared deep learning models with more traditional methods and reported considerable performance increases by adopting deep learning architectures over more traditional learners. Yet, the authors used a methodology which relied upon simple TF-IDF vector representations to structure the textual input for the traditional learners. Unlike GloVe word embeddings, this representation does not capture the semantic meaning linked to words, and this could explain the underestimation of traditional methods' performance. Our findings show that ML methods are highly competitive when adequate feature engineering is performed, with only a state-of-the-art transformer

architecture (i.e., H-BERT) achieving similar predictive performance. Interestingly, we observe the relative performance of H-BERT compared to the traditional learners to increase with increased data set size (see last row in Table 4). This finding is in line with previous studies which conclude that fine-tuning transformer models is more complicated and variable for small data sets (Mosbach et al., 2020), and that traditional machine learning models are expected to outperform deep learning models when using limited amounts of data (Kraus et al., 2020). Using only the PHEME data sets would have resulted in a clear preference towards an ML algorithm (i.e., support vector machine), while larger data sets point towards the superiority of the H-BERT model, with the best performance on the largest data set (i.e., COVID pandemic). This stresses the importance of the introduction of new, larger data sets to the field.

4.2. Explanation quality

4.2.1. Fidelity

Table 5: Fidelity scores of each algorithm across all seven benchmarked data sets

	Charlie Hebdo	Ferguson Unrest	Germanwings Crash	Ottawa Shooting	Sydney Siege	COVID Pandemic	Amazon Fires	**Average Rank**
LR	**0.3531**	0.2865	0.2853	0.1827	0.3112	0.3877	**0.2810**	2.00
SVM	0.3454	**0.3265**	**0.3707**	**0.2007**	**0.3989**	**0.4220**	0.2808	**1.29**
RF	0.2186	0.2343	0.3143	0.1804	0.3471	0.3537	0.2185	2.71
H-CNN	0.0543	0.0819	0.0847	0.1004	0.0895	0.0565	0.0493	4.43
H-LSTM	0.0321	0.0507	0.0755	0.0560	0.0468	0.0492	0.0426	6.29
H-GRU	0.0310	0.0517	0.0868	0.0762	0.0478	0.0448	0.0424	6.00
H-BERT	0.0436	0.0538	0.0516	0.0551	0.0525	0.0734	0.0918	5.29
# *obs*	2,079	1,143	493	890	1,221	4,612	1,392	

Explanation fidelity results, as depicted in Table 5, lead to similar remarks as predictive accuracy results. Again, the ML models are among the top performers, while explanations generated from the deep learning models are generally not very trustworthy. The shift of H-BERT from the top predictive performers to the least trustworthy model explanations is remarkable. To understand this, it is imperative to understand the two main drivers behind explanation fidelity. First, predictive model performance is important. If the original black-box model is good at learning the relationship between predictor variables and response, this signifies that the explanation mechanism can learn *actual*

relationships rather than noise. This explains the top performance of SVM on both model performance as well as explanation fidelity. Second, explanation fidelity measures how well the explanation method approximates the actual model behavior (also called algorithm-explanation compatibility). H-BERT is among the top predictive performers, yet LIME is unable to provide meaningful explanations based on the algorithm due to large differences between our interpretable data representation $\boldsymbol{x_{INTERPRETABLE}}$ and the algorithm-fed representation $\boldsymbol{x_{LIME}}$. An attention-based representation is capable of capturing rich contextual information from texts, which may distinguish between complex cases such as homonyms, sarcasm, or negations. As Ribeiro et al. (2016) already warned for, certain interpretable representations (such as a term frequency vector) might not be capable of deriving such information and hence might not be powerful enough to explain the behavior of complex black box models. Our results seem to empirically validate these claims. Interestingly, GloVe word embeddings, which also capture semantic information, seem more compatible with a simple human-interpretable term frequency representation. This results into much higher fidelity scores, as they entail a reduced level of complexity when compared to the H-BERT architecture. Thus, relationships learned in the fitted H-BERT model may be impossible to mimic using interpretable data representations. A similar pattern can be envisioned for the other DL models, which aim to interpret the sequential nature of the text, which is unaccounted for in the LIME methodology. This signifies that explanation methods like LIME, which are currently deemed model-agnostic, may no longer be truly model-agnostic due to the ever-increasing complexity observed in deep learning architectures. Whereas this limitation was already mentioned by Ribeiro et al. (2016), we are the first to underpin this statement with explanation evaluation metrics and to show the differential impact across different deep learning methods.

Figure 5: Example Tweet

To clarify the difference between ML and DL in terms of interpretation, we provide an example of an explanation on 1 local instance for 1 fold using LIME. Figure 5 depicts an example of an actual COVID-19 tweet which was labeled as a non-rumor, while Table 6 contains the detailed results when deploying LIME for this example on two trained algorithms (i.e., SVM and H-BERT). When deploying LIME on our trained SVM (Table 6, panel A), LIME pointed towards the usage of the word 'input' and the tag '@ScienceMagazine', combined with the follower ratio of the account, as being indicative of

having a low probability of being a rumor. This is highly human-interpretable and informative. The word 'input' indicates some form of collaboration, which means that at least two entities should be formulating this expression, lowering the possibility of a non-justified claim. Science Magazine is one of the most respectable scientific journals, which also lowers the possibility of unfounded claims, and the follower ratio points towards the online presence of a respected figure of the scientific community. This heavily differs from the results when deploying LIME on the H-BERT model (Table 6, panel B). While this model also performs well, the three LIME coefficients with the highest absolute value are much less informative: the use of the word 'an', the use of a third person verb (VBZ), and the absence of 'where' or 'when' (WRB). This explanation looks rather non-sensical from a human perspective. It seems that LIME is unable to uncover the underlying relationship between language use (i.e., through individual words) and predicted probability. Nonetheless, language use should be learned by H-BERT as it clearly outperforms the other DL models on the COVID data set, and the sole architecture distinction with these other DL methods is the handling of the unstructured textual features, as the structured features are fed through an identical feed-forward part of the architecture in all DL models in our approach. This inability to uncover the underlying fitted model is also identified by the fidelity scores of the white box explanatory model, with a score of 0.39 for the SVM model, and a much lower score of 0.06 for the H-BERT model. This is especially worrisome as LIME is often used to explain complicated text models which use transformer architectures (e.g., Stevenson et al. (2021)). Future academic research, and practitioners, should be extremely cautious before doing so, and use adequate explanation quality evaluation methods before interpreting the results, such as the one proposed in this study.

Table 6: Top-10 Features SVM (panel A) and H-BERT (panel B). Term Frequency Features indicated in bold and underlined.

Panel A		Panel B	
SVM Feature	Standardized Coefficient Value	H-BERT Feature	Standardized Coefficient Value
input	-0.0195	**an**	-0.0255
Follower Ratio	-0.0119	VBZ	-0.0181
sciencemagazine	-0.0092	WRB	-0.0140
IN	-0.0074	DT	-0.0068
CD	-0.0065	PRP	-0.0060
by	-0.0056	CC	0.0058
behave	-0.0052	VB	-0.0057
Word Count	-0.0048	FW	0.0057
epidemics	-0.0044	PRP$	-0.0044
Listed Count	-0.0034	Period	-0.0044

4.2.2. Stability

Table 7: Explanation stability scores ($Stability_{10}$) of each algorithm across all seven benchmarked data sets. Minimum and maximum pairwise Jaccard values indicated between brackets.

	Charlie Hebdo	Ferguson Unrest	Germanwings Crash	Ottawa Shooting	Sydney Siege	COVID Pandemic	Amazon Fires	**Average Rank**
LR	0.1004	0.0875	0.1232	0.0920	0.0921	0.1008	0.0918	5.71
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	(0.3333)	(0.2500)	(0.3333)	(0.4286)	(0.3333)	(0.2500)	(0.4286)	
SVM	0.1063	0.0943	0.1208	0.1112	0.0814	0.0898	0.0690	6.14
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	(0.4286)	(0.4286)	(0.2500)	(0.2500)	(0.2500)	(0.2500)	(0.2500)	
RF	0.0985	0.1207	0.1273	0.1240	0.0866	0.0696	0.0852	5.57
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	(0.4286)	(0.4286)	(0.3333)	(0.2500)	(0.3333)	(0.1765)	(0.2500)	
H-CNN	**0.1693**	0.1034	**0.1734**	0.1681	0.1229	0.0881	0.1087	3.14
	(0.0526)	(0.0526)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	(0.3333)	(0.3333)	(0.4286)	(0.3333)	(0.2500)	(0.4286)	(0.3333)	
H-LSTM	0.1326	0.1345	0.1571	0.1222	0.1427	0.1135	0.1284	3.00
	(0.0526)	(0.0526)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	(0.3333)	(0.3333)	(0.4286)	(0.3333)	(0.2500)	(0.4286)	(0.3333)	
H-GRU	0.1400	0.1294	0.1714	0.1650	0.1370	0.1189	0.1044	2.86
	(0.0526)	(0.0526)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	(0.3333)	(0.3333)	(0.4286)	(0.3333)	(0.2500)	(0.4286)	(0.3333)	
H-BERT	0.1474	**0.1474**	0.1423	**0.1706**	**0.1650**	**0.1443**	**0.1410**	**1.57**
	(0.0526)	(0.0526)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)	
	(0.3333)	(0.3333)	(0.4286)	(0.3333)	(0.2500)	(0.4286)	(0.3333)	
# *obs*	2,079	1,143	493	890	1,221	4,612	1,392	

A very different conclusion can be made when looking at the explanation stability results in Table 7. Deep learning models consistently give more stable global explanations compared to traditional learners. However, their explanations are not trustworthy, as elaborated in the previous paragraph. It seems that these models rather give 'generic' explanations that are similar across folds but do not represent the actual modeled relationships. This is also suggested by the minimum and maximum values that vary minimally across algorithms for these data sets. The ML methods are far more unstable, and this issue is even aggravated when interpreting the metric. A stability value of 0.33 would correspond to an average of 50% agreement across folds (see 3.4. Model evaluation). We only reach a maximum

stability value of 0.1734 or about 30% agreement across folds, and the ML algorithms achieve stability scores of 0.10 or lower. This means that if practitioners or academics would decide to interpret feature importances based upon one fold, changing the fold used for interpretation would alter more than half of the features that are deemed important. Such instability could heavily hamper the widespread acceptance of decision support systems, as a global model perspective is important to ascertain trust in the model (Ribeiro et al., 2016). Lower trust could then result into lower adoption rates of such systems, further allowing the widespread use of unverified rumors. Moreover, a similar observation can be made when looking at the results for the stability of the top-5 or top-20 most important global features, as reported in Appendix F. This widespread instability indicates that the evaluation of explanation quality is essential before reporting explanations. Our results indicate that the resulting global explanations are highly unstable and that the local explanations vary heavily in fidelity based upon the type of model, with a better fit towards ML models than DL architectures. Overall, it seems that high-dimensional human-interpretable data representations often lead to unstable global explanations, which is especially worrisome given the importance of these global explanations with relation to trust and acceptance of these methods.

One could also argue that such unstable global explanations cast doubt upon the quality of the local explanations. However, we argue that this is not necessarily the case, and that unstable global explanations are in fact the result of powerful local explanations that are able to detect the influence of individual words on the predicted outcome for individual observations. To make this clearer, we compare a high-accuracy, high-fidelity, and low-stability SVM model with a low-accuracy, low-fidelity, and high-stability H-CNN model trained on the Charlie Hebdo data set. The top-10 most important features across the various folds are reported in Appendix G. For H-CNN, LIME is not capable of detecting the influence of individual words, given that individual words are only detected as important features once in seven out of the ten folds, representing only 7% of all important features across folds. On the other hand, LIME on SVM is capable of deriving actual influence from individual words. This is reflected in a much larger presence of individual words in the most important features, with 22% of all important features across folds being words. We note that the inclusion of individual words drastically increases the number of candidate important features up to hundreds of features, while there are only a couple dozen of structured features. Moreover, the individual word features are also very sparse, with most words only occurring in one or a few individual tweets (Appendix H visualizes the frequency distribution of the 300 most common words of each data set which shows how quickly these words become sparse). This makes it harder for these features (which may be very important per individual explanation) to extrapolate to the aggregated global explanation and be picked as important feature for the entire data set, especially across the folds. Thus, the fact that the term frequency-based representation used for model explanation is incapable of trustworthy mimicking DL model behavior, which uses more complicated textual representations (see 4.2.1. Fidelity), results into ‘falsely stable’ global explanations.

This is even aggravated when we perform a small robustness check. For this sample case, we built a 'non-hybrid' CNN model (i.e., only using sequential textual inputs), and inspected how the stability scores changed when the model no longer had the option to select the structured features. The stability scores dropped remarkably to low values ($Stability_{20}$ = 0.0581, $Stability_{10}$ = 0.0467, $Stability_{5}$ = 0.0370). The underlying top-10 features per fold are reported in Appendix Table G.3. Hence, when only textual features are used, DL models are shown to also result into highly unstable global explanations. This may be the best evidence that the DL-based LIME explanations are not trustworthy and that their perceived stability is induced by false explanations. In the example for the Charlie Hebdo data set mentioned above, the CNN-based explanations seem to draw randomly from the limited set of structured features, while the SVM-based explanations use the much larger full potential set of structured and unstructured features, which results in more unstable global explanations.

4.3. Trade-off

The aforementioned results hint towards three types of outcomes: (1) ML models in the high predictive accuracy, high fidelity category, and the deep learners always achieving low fidelity, either with (2) high or (3) low predictive accuracy. To check this, we input the results as displayed in Tables 4, 5, and 7 into a K-means clustering algorithm (Lloyd, 1982), with each metric (i.e., AUC, fidelity and stability) acting as one dimension of a three-dimensional clustering space. The results with K equal to 3 are displayed in Figure 6. The algorithm generated three clearly distinct clusters: the blue cluster depicts the high fidelity algorithms, and the green and orange clusters separate the low fidelity cluster through respectively low and high predictive performance. We notice that the high fidelity cluster is formed solely by ML methods, with the random forest and logistic regression fitted on the Ottawa shooting data set being the sole two ML models not present in this cluster. Interestingly, these two ML models are the two orange dots located very closely to the high fidelity cluster on panels A and C. Panels A and C clearly visualize how the blue cluster (i.e., ML models) is the only segment capable of reaching a relatively high fidelity. However, there does not seem to be a relationship between fidelity and predictive accuracy (i.e., AUC) as the predictive performance of these methods is high in general. This may not be the case for other tasks where deep learning outperforms traditional learners, such as computer vision (Krizhevsky, Sutskever & Hinton, 2017), where LIME is also commonly applied (Meske & Bunde, 2020). However, our results clearly indicate that the generated explanations are untrustworthy, and that there is a clear trade-off between LIME fidelity and complexity of original model (i.e., the more complex DL models show lower fidelity). Also in rumor identification, one could argue to adopt transformer architectures when data set sizes would keep growing. Nonetheless, users should be aware of the fact that such complex models cannot easily be explained using model-agnostic techniques such as LIME and that adequate explanation evaluation is required.

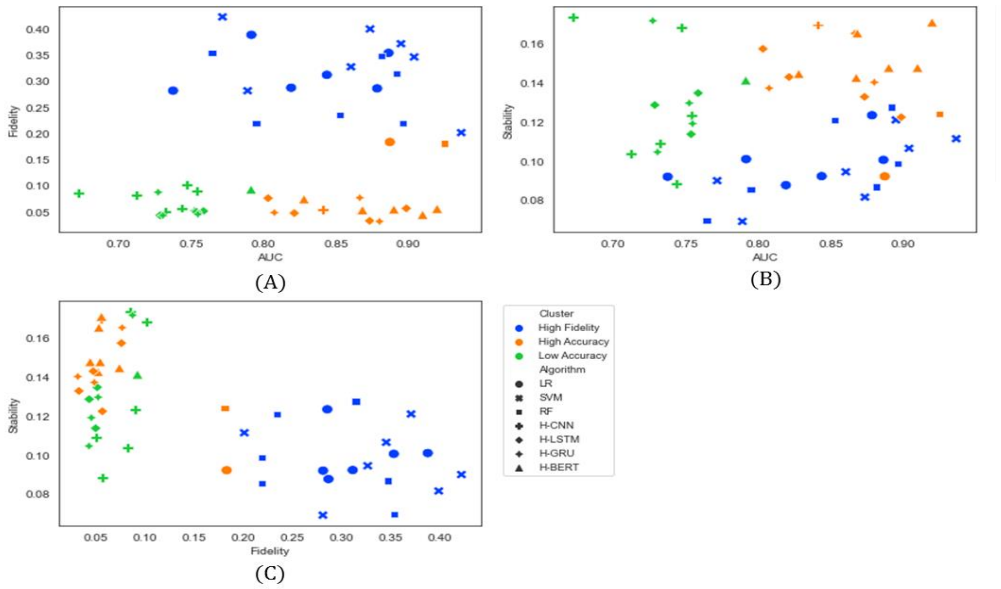

Figure 6: Visualization relationship AUC-Fidelity-Stability

Looking at panel B, no clear relationship seems to be present between AUC and stability, but there seems to be a trade-off between fidelity and stability (panel C), where locally trustworthy explanations are not capable of creating stable global explanations. These results are in line with the robustness checks performed in Section 4.2., where the DL models are shown to only have 'apparent' stability due to the usage of untrustworthy explanations. This is unfortunate given that, ideally, trustworthy explanations are also globally stable, and that LIME is currently also being aggregated to formulate global model explanations in the OR domain (e.g., Stevenson et al. (2021)). Moreover, instability is also observed at the local level for model-agnostic perturbation techniques such as LIME and SHAP in other studies (e.g., Alvarez-Melis & Jaakkola (2018)).

5. Discussion and managerial implications

This study aims to answer several research questions. A first research question focuses on which rumor detection model performs best when including both unstructured and structured features. Our results indicate that traditional machine learning methods and transformer architectures have the highest performance when evaluating solely on predictive performance, while only the traditional learners also translate this high predictive performance into quality explanations. The better performance of the BERT model compared to other DL metrics is interesting in the light of a previous study on rumor detection (Almars et al., 2022). The authors demonstrate the superiority of an attention-based model, but also report very weak performance from a pre-trained BERT model. However, the application was in Arabic, which means that another pre-trained model was used and no information on the model fine-tuning is provided.

The top performance of traditional learners contradicts previous studies focusing on predictive accuracy. Past research (Kumar et al., 2020) suggests the superiority of deep learning architectures over traditional learners in the context of rumor detection. However, this might be due to the fact that they solely used threshold-dependent performance metrics to compare deep and traditional classifiers. Table 8 and 9 show this by focusing on the predictive performance of the imbalanced data sets (Amazon bushfires and COVID19) using threshold-dependent metrics (i.e., accuracy, precision, recall, and F1-measure). The imbalanced sets were selected as a previous comparative study between ML and DL classifiers for rumor detection combined the PHEME data sets into one large data set which had minor class imbalance with 34% rumors and 66% non-rumors. As demonstrated below, class imbalance can heavily impact accuracy results. The classification threshold was set equal to the proportion of the minority class in the training fold, a popular method when the misclassification costs are unknown (Bequé & Lessmann, 2017). Most importantly, when using the F1 measure, a popular measure for imbalanced classification tasks (Mahajan et al., 2020), the conclusions with regard to the preferred algorithm remain unchanged (i.e., ML and BERT outperforming the other DL models). However, the results clearly show how these metrics could lead to incorrect conclusions. For instance, focusing on ML models in the Amazon data set (Table 8), the F1 measure and accuracy both indicate the random forest as the weakest classifier among the ML models. However, the low precision and high recall simply indicate that the classifier tends to output higher probabilities, which leads to a higher number of estimated rumors. This problem is clearly threshold-dependent, thereby further validating our adoption of a threshold-independent metric, away from current practice in rumor detection literature. Moreover, the threshold-dependent metrics still favor the ML and BERT models, as they have the best F1 scores and are the only models capable of obtaining relevant recall scores, thereby effectively hindering the spread of unverified information. In fact, the results in Tables 8 and 9 are yet another evidence of the dangers of using metrics such as accuracy, as they result into much better scores for the DL architectures that use recurrent or convolutional layers, while a deeper interpretation clearly suggests a better performance from the ML models. It might in fact be that this focus on accuracy has resulted into the outperformance of DL models compared to ML models (e.g., Kumar et al., 2020).

Table 8: Median Performance Scores Alternative Metrics on Amazon Bushfires Data Set

	ACC	PREC	REC	F1
LR	0.7953	0.3407	0.5419	0.4120
SVM	0.7737	0.3291	0.6688	**0.4431**
RF	0.6329	0.2368	**0.7934**	0.3578
H-CNN	**0.8671**	0.4393	0.2121	0.3000

H-LSTM	0.8628	**0.4611**	0.3352	0.3890
H-GRU	0.8563	0.4289	0.3151	0.3569
H-BERT	0.7234	0.2866	0.7152	0.4139

Table 9: Median Performance Scores Alternative Metrics on COVID19 Data Set

	ACC	PREC	REC	F1
LR	0.7706	0.2614	0.6317	0.3700
SVM	0.7279	0.2134	0.7051	0.3288
RF	0.5863	0.1775	**0.8103**	0.2912
H-CNN	**0.8877**	**0.4327**	0.1846	0.2522
H-LSTM	0.8721	0.3895	0.3113	0.3418
H-GRU	0.8721	0.3925	0.3347	0.3432
H-BERT	0.8643	0.2552	0.5347	**0.3801**

To rule out any other impact of class imbalance in the novel data sets on the results, we inspect how different levels of oversampling the minority class influences predictive performance in Figure 7. Besides the traditional 50/50 resampling, we also perform a limited oversampling (i.e., a 25/75 distribution) as this setting preserves the imbalance between rumors and non-rumors and this was previously observed to result into optimal predictive performance (Del Rio, Benítez & Herrera, 2015). Figure 7 clearly shows a limited influence from resampling on the overall results. For each of the settings, DL methods (besides BERT) fail to reach competitive predictive performance compared to ML and BERT. The plots also indicate that by applying oversampling the performance of most algorithms decreases. A possible explanation might be that given the limited sample sizes oversampling will heavily overweight minority instances, thereby making certain classifiers more prone to overfitting on these observations, which on its part decreases the generalizability of these classifiers and eventually degrades their performance.

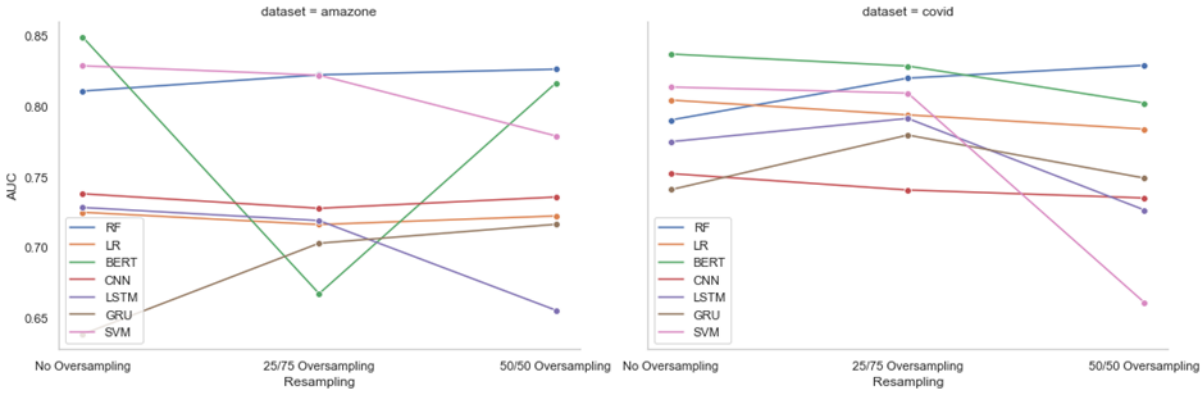

Figure 7: Predictive Performance with and without oversampling on one fold of imbalanced data sets (Amazon and Covid). Techniques were deployed on one random fold used in main analysis.

The second research question focuses on the development of a methodology to effectively measure the quality of the explanations and, related, which type of model has the highest explanation quality. A novel methodology is suggested to measure explanation quality with regard to both fidelity and stability. Fidelity leverages local model goodness-of-fit, while global stability is measured through an adapted Jaccard score which is applied to the top-N aggregated coefficient values across folds. We note that the developed stability metric is widely applicable, as it only requires importance rankings and is agnostic about the source of these rankings (e.g., permutation vs variance-based). As discussed above, this argues towards the selection of traditional learners over deep learners, further counter arguing against the predictive superiority of deep learners (Kumar et al., 2020). Moreover, it shows that all models are highly unstable with different global model explanations across folds, which makes it more difficult to get the models accepted by business users.

Finally, the study also investigates whether a trade-off exists between predictive performance and explanation quality. For our study, this seems not to be the case. However, our results clearly suggest a trade-off between explanation quality and model complexity. This means that model-agnostic explanation techniques deliver explanations of lower quality when deployed on more complex learners, despite the widespread use of these explanation techniques on complex learners (Zinovyeva et al., 2020; Stevenson et al., 2021). Interestingly, this low performance on explanation quality can be explained through the larger discrepancy between algorithm representation and interpretable representation on word level, as currently deployed in academic literature (Zinovyeva et al., 2020; Stevenson et al., 2021). With the growing popularity of complex large language models, it might be interesting to develop a humanly interpretable representation which is more reminiscent of the underlying text representations used in these models, which might make them more compatible with model-agnostic explanation methods.

The introduction of new data sets has the potential to challenge scholars further in the domain. When inspecting Table 4, we observe lower AUC values for the new imbalanced COVID-19 and Amazon bushfire data sets compared to the equally distributed PHEME data sets. This is surprising as,

theoretically, AUC scores should be enhanced when class imbalance increases. It suggests that the newly introduced data sets are much harder to predict compared to the PHEME data sets, which are currently widely used in the field. This further enhances the usefulness of the new data sets.

Managers and practitioners should take note of these outcomes as they have several important implications. First of all, the past evidence favoring deep learning models seems to be contradicted, and practitioners should be cautious when deploying such models. An overall evaluation across algorithms seems to be important, and this evaluation should be ideally done using threshold-independent metrics when misclassification costs are unknown. Especially accuracy seems to give biased outcomes. Moreover, this evaluation should also focus on the quality of algorithm explanations. Currently deemed model-agnostic explanation techniques are proven to be not truly model-agnostic. Several recent papers (Zinovyeva et al., 2020; Stevenson et al., 2021) deploy these techniques without an evaluation of their quality, which could result in using untrustworthy local explanations and incorrect business decisions being made. Finally, global interpretations of a model prove to be highly unstable across folds, meaning users should be careful when presenting interpretations based on a single fold. We therefore advise to report stability measures together with the interpretations.

6. Conclusions and future research

This study contributes to rumor detection literature with a specific focus on (1) combining structured and unstructured features and (2) evaluating model explanation quality. Moreover, it introduced two new data sets about major recent disasters (i.e., COVID-19 pandemic, and Amazon bushfires), and combined these with the popular PHEME data sets to set-up a benchmark study. The study evaluates the predictive performance of several algorithms, as well as their quality of their corresponding explanations through LIME. Both fidelity and stability of explanations are evaluated. The results are especially positive towards traditional machine learning methods (i.e., support vector machine, logistic regression, and random forest) which achieve both high predictive accuracy and trustworthy local explanations through the LIME framework. Most deep learning architectures, on the other hand, score low on predictive performance and model explainability, with the clear exception being H-BERT. Finally, all algorithms resulted into global explanations that were either highly unstable across folds (i.e., ML models) or were falsely considered stable due to untrustworthy local explanations (i.e., DL models). The results thus urge researchers and practitioners to be very cautious when providing model interpretations based on one fold or sample.

The detection of these untrustworthy and unstable explanations was possible thanks to an explanation quality evaluation methodology introduced in this study. The method validates both local fidelity as well as global stability. The results clearly indicate issues with LIME when used for global explanations, as well as when LIME is used for local explanations of deep learning models.

There seems to be no trade-off present between predictive performance and explanation quality in our case. However, there seems to be a trade-off between fidelity and model complexity. This

indicates that there could be a trade-off between predictive performance and explanation quality in fields where complex models thrive. Fortunately, this is typically not the case for OR applications (Kraus et al., 2020).

Since we are the first to conduct such an elaborate benchmark study for explanation quality in rumor detection, we decided to focus on one popular explanation method. However, future research may validate whether similar conclusions can be formulated for other popular model-agnostic perturbation techniques such as SHAP, and whether similar results could be found in other research contexts, which may not be characterized by the high-dimensional unstructured features as present in rumor detection. Moreover, future research may look into interpretable meta-features (e.g., topics) that could replace individual words in order to create more stable text-based interpretations.

Social media posts encapsulate more than simply text. Visual keys such as images or emojis also contain information and could aid in detecting unverified rumors. However, as we wanted to focus on textual detection methods and their explanations, we did not consider this information in our study. However, it might be interesting to investigate how these visual cues can improve rumor detection and how they would be ideally explained, as well as the quality of these explanations would be.

Finally, the study focuses on English-language data sets which are generally small. Some non-English language data sets have been provided (Alqurashi et al., 2021; Al-Sarem et al., 2021) that are larger in size. It would be extremely interesting to see if similar conclusions could be formulated on these data sets. To further validate our results, larger English data sets are necessary as well, although it is difficult to provide these on a sufficient scale if the same rigorous annotation guidelines are followed as in Zubiaga et al. (2015) and in our study.

Acknowledgements

This research is funded by the Ghent University's Special Research Fund (BOF) [BOF/STA/202009/001] and the Research Foundation Flanders' (FWO) postdoctoral fellowship [12ZM923N].

References

Ajao, O., Bhowmik, D., & Zargari, S. (2018, July). 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).

Al-Sarem, M., Boulila, W., Al-Harby, M., Qadir, J., & Alsaeedi, A. (2019). Deep learning-based rumor detection on microblogging platforms: a systematic review. IEEE Access, 7, 152788-152812.

Al-Sarem, M., Alsaeedi, A., Saeed, F., Boulila, W., & AmeerBakhsh, O. (2021). A novel hybrid deep learning model for detecting COVID-19-related rumors on social media based on LSTM and concatenated parallel CNNs. Applied Sciences, 11(17), 7940.

Almars, A. M., Almaliki, M., Noor, T. H., Alwateer, M. M., & Atlam, E. (2022). HANN: Hybrid Attention Neural Network for Detecting Covid-19 Related Rumors. IEEE Access, 10, 12334-12344.

Alqurashi, S., Hamoui, B., Alashaikh, A., Alhindi, A., & Alanazi, E. (2021). Eating garlic prevents COVID-19 infection: Detecting misinformation on the Arabic content of Twitter. arXiv preprint arXiv:2101.05626.

Alvarez-Melis, D., & Jaakkola, T. S. (2018). On the robustness of interpretability methods. arXiv preprint arXiv:1806.08049.

Ballings, M., & Van den Poel, D. (2015). CRM in social media: Predicting increases in Facebook usage frequency. European Journal of Operational Research, 244(1), 248-260.

Bastos, J. A., & Matos, S. M. (2022). Explainable models of credit losses. European Journal of Operational Research, 301(1), 386-394.

Bequé, A., & Lessmann, S. (2017). Extreme learning machines for credit scoring: An empirical evaluation. Expert Systems with Applications, 86, 42-53.

Bondielli, A., & Marcelloni, F. (2019). A survey on fake news and rumour detection techniques. Information Sciences, 497, 38-55.

Borchert, P., Coussement, K., De Caigny, A., & De Weerdt, J. (2022). Extending business failure prediction models with textual website content using deep learning. European Journal of Operational Research.

Bücker, M., Szepannek, G., Gosiewska, A., & Biecek, P. (2022). Transparency, auditability, and explainability of machine learning models in credit scoring. Journal of the Operational Research Society, 73(1), 70-90.

Coussement, K., & Benoit, D. F. (2021). Interpretable data science for decision making. Decision Support Systems, 150, 113664.

Chen, Y. C., Liu, Z. Y., & Kao, H. Y. (2017, August). Ikm at semeval-2017 task 8: Convolutional neural networks for stance detection and rumor verification. In Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)(pp. 465-469).

Cinelli, M., Quattrociocchi, W., Galeazzi, A., Valensise, C. M., Brugnoli, E., Schmidt, A. L., ... & Scala, A. (2020). The covid-19 social media infodemic. Scientific Reports, 10(1), 1-10.

Del Rio, S., Benítez, J. M., & Herrera, F. (2015, August). Analysis of data preprocessing increasing the oversampling ratio for extremely imbalanced big data classification. In 2015 IEEE Trustcom/BigDataSE/ISPA (Vol. 2, pp. 180-185). IEEE.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. Neural computation, 10(7), 1895-1923.

DiFonzo, N., & Bordia, P. (2007). Rumor, gossip and urban legends. Diogenes, 54(1), 19-35.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), 654-669.

Hamidian, S., & Diab, M. T. (2019). Rumor detection and classification for twitter data. arXiv preprint arXiv:1912.08926.

Janssens, B., Bogaert, M., & Maton, M. (2022). Predicting the next Pogačar: a data analytical approach to detect young professional cycling talents. Annals of Operations Research, 1-32.

Ke, L., Chen, X., Lu, Z., Su, H., & Wang, H. (2020, October). A Novel Approach for Cantonese Rumor Detection based on Deep Neural Network. In 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 1610-1615). IEEE.

Kim, Y., Kim, H. K., Kim, H., & Hong, J. B. (2020). Do many models make light work? evaluating ensemble solutions for improved rumor detection. IEEE Access, 8, 150709-150724.

Kochkina, E., Liakata, M., & Zubiaga, A. (2018). All-in-one: Multi-task learning for rumour verification. arXiv preprint arXiv:1806.03713.

Kraus, M., Feuerriegel, S., & Oztekin, A. (2020). Deep learning in business analytics and operations research: Models, applications and managerial implications. European Journal of Operational Research, 281(3), 628-641.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.

Kumar, A., Singh, V., Ali, T., Pal, S., & Singh, J. (2020). Empirical Evaluation of Shallow and Deep Classifiers for Rumor Detection. In Advances in Computing and Intelligent Systems (pp. 239-252). Springer, Singapore.

Le, Q., & Mikolov, T. (2014, June). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196). PMLR.

Liu, X., Nourbakhsh, A., Li, Q., Fang, R., & Shah, S. (2015, October). Real-time rumor debunking on twitter. In Proceedings of the 24th ACM international on conference on information and knowledge management (pp. 1867-1870).

Lloyd, S. (1982). Least squares quantization in PCM. IEEE transactions on information theory, 28(2), 129-137.

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. Advances in neural information processing systems, 30.

Meire, M., Hewett, K., Ballings, M., Kumar, V., & Van den Poel, D. (2019). The role of marketer-generated content in customer engagement marketing. Journal of Marketing, 83(6), 21-42.

Martens, D., Baesens, B., Van Gestel, T., & Vanthienen, J. (2007). Comprehensible credit scoring models using rule extraction from support vector machines. European journal of operational research, 183(3), 1466-1476.

Meske, C., & Bunde, E. (2020, July). Transparency and trust in human-AI-interaction: The role of model-agnostic explanations in computer vision-based decision support. In International Conference on Human-Computer Interaction (pp. 54-69). Springer, Cham.

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. Artificial Intelligence, 267, 1-38.

Molnar, C. (2022). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable (2nd ed.). christophm.github.io/interpretable-ml-book/

Mosbach, M., Andriushchenko, M., & Klakow, D. (2020). On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. *arXiv preprint arXiv:2006.04884*.

Mridha, M. F., Keya, A. J., Hamid, M. A., Monowar, M. M., & Rahman, M. S. (2021). A comprehensive review on fake news detection with deep learning. IEEE Access, 9, 156151-156170.

Paka, W. S., Bansal, R., Kaushik, A., Sengupta, S., & Chakraborty, T. (2021). Cross-SEAN: A cross-stitch semi-supervised neural attention model for COVID-19 fake news detection. Applied Soft Computing, 107, 107393.

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

Peters, M. E., Ruder, S., & Smith, N. A. (2019). To tune or not to tune? Adapting pretrained representations to diverse tasks. *arXiv preprint arXiv:1903.05987*.

Qazvinian, V., Rosengren, E., Radev, D., & Mei, Q. (2011, July). Rumor has it: Identifying misinformation in microblogs. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (pp. 1589-1599).

Ramon, Y., Martens, D., Evgeniou, T., & Praet, S. (2021). Can metafeatures help improve explanations of prediction models when using behavioral and textual data?. Machine Learning, 1-40.

Reisach, U. (2021). The responsibility of social media in times of societal and political manipulation. European Journal of Operational Research, 291(3), 906-917.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " 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).

Stevenson, M., Mues, C., & Bravo, C. (2021). The value of text for small business default prediction: A deep learning approach. European Journal of Operational Research, 295(2), 758-771.

Searle, J. R. (1975). A taxonomy of illocutionary acts.

Shankaranarayana, S. M., & Runje, D. (2019, November). ALIME: Autoencoder based approach for local interpretability. In International conference on intelligent data engineering and automated learning (pp. 454-463). Springer, Cham.

Stieglitz, S., Dang-Xuan, L., Bruns, A., & Neuberger, C. (2014). Social media analytics-an interdisciplinary approach and its implications for information systems. Business & Information Systems Engineering, 6(2), 89-96.

Theil, M. (2022) A history of the Royal Family's Twitter account, from sending the Queen's first tweet to announcing her death. https://www.insider.com/history-royal-family-twitter-queen-announcing-death-2022-9 (Accessed Oct 19, 2022)

Van Belle, J., Crevits, R., & Verbeke, W. (2022). Improving forecast stability using deep learning. International Journal of Forecasting.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.

Visani, G., Bagli, E., Chesani, F., Poluzzi, A., & Capuzzo, D. (2022). Statistical stability indices for LIME: Obtaining reliable explanations for machine learning models. Journal of the Operational Research Society, 73(1), 91-101.

Yang, Z., Wang, C., Zhang, F., Zhang, Y., & Zhang, H. (2015, September). Emerging rumor identification for social media with hot topic detection. In 2015 12th web information system and application conference (WISA) (pp. 53-58). IEEE.

Yu, F., Liu, Q., Wu, S., Wang, L., & Tan, T. (2017, August). A Convolutional Approach for Misinformation Identification. In IJCAI (pp. 3901-3907).

Yuan, H., Zheng, J., Ye, Q., Qian, Y., & Zhang, Y. (2021). Improving fake news detection with domain-adversarial and graph-attention neural network. Decision Support Systems, 151, 113633.

Zhang, C., Gupta, A., Kauten, C., Deokar, A. V., & Qin, X. (2019). Detecting fake news for reducing misinformation risks using analytics approaches. European Journal of Operational Research, 279(3), 1036-1052.

Zhang, T., Wu, F., Katiyar, A., Weinberger, K. Q., & Artzi, Y. (2020). Revisiting few-sample BERT fine-tuning. *arXiv preprint arXiv:2006.05987*.

Zubiaga, A., Liakata, M., Procter, R., Wong Sak Hoi, G., & Tolmie, P. (2016). Analysing how people orient to and spread rumours in social media by looking at conversational threads. PloS one, 11(3), e0150989.

Zubiaga, A., Liakata, M., & Procter, R. (2017). Exploiting context for rumour detection in social media. In International Conference on Social Informatics (pp. 109-123). Springer, Cham.

Zinovyeva, E., Härdle, W. K., & Lessmann, S. (2020). Antisocial online behavior detection using deep learning. Decision Support Systems, 138, 113362.