A Language-Based Approach to Fake News Detection Through Interpretable Features and BRNN

Qiao, Y.; Wiechmann, D.; Kerz, E.

Publication date
2020

Document Version
Final published version

Published in
3rd International Workshop on Rumours and Deception in Social Media (RDSM)

A Language-Based Approach to Fake News Detection Through Interpretable Features and BRNN

Yu Qiao[1], Daniel Wiechmann[2], Elma Kerz[1*]
RWTH Aachen University[1], University of Amsterdam[2]
yu.qiao@rwth-aachen.de,d.wiechmann@uva.nl
elma.kerz@ifaar.rwth-aachen.de

Abstract

'Fake news' – succinctly defined as false or misleading information masquerading as legitimate news – is a ubiquitous phenomenon and its dissemination weakens the fact-based reporting of the established news industry, making it harder for political actors, authorities, media and citizens to obtain a reliable picture. State-of-the art language-based approaches to fake news detection that reach high classification accuracy typically rely on black box models based on word embeddings. At the same time, there are increasing calls for moving away from black-box models towards white-box (explainable) models for critical industries such as healthcare, finances, military and news industry. In this paper we performed a series of experiments where bi-directional recurrent neural network classification models were trained on interpretable features derived from multi-disciplinary integrated approaches to language. We apply our approach to two benchmark datasets. We demonstrate that our approach is promising as it achieves similar results on these two datasets as the best performing black box models reported in the literature. In a second step we report on ablation experiments geared towards assessing the relative importance of the human-interpretable features in distinguishing fake news from real news.

1 Introduction

The topic of 'disinformation' – an umbrella term used to encompass a wide range of types of information disorder, "including 'fake news', rumors, deliberately factually incorrect information, inadvertently factually incorrect information, politically slanted information, and 'hyperpartisan' news" (Tucker et al., 2018) – is attracting more and more attention. This reflects a deeper concern that the prevalence of disinformation leads to an increased political polarization, decreases trust in public institutions, and undermines democracy. For example, the spread of 'fake news' – concisely defined as intentionally false information masquerading as genuine news – for financial and political gains had a potential impact on the contentious Brexit referendum or 2016 U.S. presidential elections (Allcott and Gentzkow, 2017; Ward, 2018). Against this background, it is hardly surprising that there has been an increased interest in the development of methods, measures and computational tools that efficiently and effectively detect disinformation using machine learning and deep learning techniques. Among different approaches to fake news detection, language-based approaches have emerged as promising (for more details, see Section 2). Here the term 'language-based' is used in a broad sense to include a variety of approaches, such as those that employ traditional linguistic features, readibility features, style-based features, discourse and rhetorical features or those that draw on word embedding techniques. The latter have proven to be particularly successful in detecting fake news. Despite their success, however, their detection is based on latent features that are not human interpretable and thus cannot explain why a piece of news was detected as fake news. As recently pointed out by Shu et al. (2019), white-box (explainable) approaches to fake news detection are desirable, since model-derived explanations can (1) provide valuable insights originally hidden to different stakeholders, such as policy makers, professional journalists and citizens and (2) can contribute to further improvement of fake news detection systems. This paper seeks to respond to recent calls for more explainable (white-box) approaches to fake news detection by performing a series

* Corresponding author

Proceedings of the 3rd International Workshop on Rumours and Deception in Social Media (RDSM), pages 14–31
Barcelona, Spain (Online), December 12, 2020.

of experiments where bi-directional recurrent neural network classifiers were trained on interpretable features derived from multi-disciplinary integrated approaches to language. The data come from two benchmark datasets and fake news detection is formulated as a binary classification and as a multiclass classification tasks correspondingly. The results of our experiments are promising, as our classification models achieve similar performance as the best-performing black box models reported in the literature. In a second step we report on ablation experiments geared towards assessing the relative importance of the human-interpretable features in distinguishing fake news from real news. The remainder of the paper is organized as follows: After a concise overview of related work in Section 2, Section 3 introduces the two data sets, Section 4 describes our approach to automated text analysis and six groups of language features used in the paper, Section 5 describes the model architecture, the training procedure and the method used to assess the relative feature importance. Sections 6 presents and discusses the main results and concluding remarks follow in Section 7.

2 Related Work

Here we provide a concise overview of recent approaches geared towards fake news detection that employ machine learning and deep learning techniques and we focus in particular on language-based approaches that are most pertinent to the purposes of this paper (for a more systematic and comprehensive overviews, see recent reviews and surveys by Shu et al. (2018), Oshikawa et al. (2020), Zhang and Ghorbani (2020) and Zhou and Zafarani (2020). Fake news detection is most often formulated as a binary classification task. However, categorizing all the news into two classes (fake vs real) is not the only conceivable way, since there are cases where the news is partially real and partially fake. A common practice is to add more classes distinguishing between several degrees of truthfulness and thus formulating fake news detection as a multi-class classification task. As will become evident later in this paper, we apply our approach to both scenarios. Three approaches to fake news detection frequently described in the literature are: (1) knowledge-based fake news detection (commonly using techniques from information retrieval to determine the veracity/truthfulness of news), (2) language-based fake news detection (drawing on traditional linguistic, style-related, readability or rhetorical features or using word embedding methods to distinguish between fake and real news) and (3) propagation-based fake news detection (typically using network analyses to determine the credibility of news sources at various stages, being created, published online and their spread via social media). Compared to knowledge-based and propagation-based approaches, language-based approaches are advantageous for several reasons, including: (1) they enable near real-time feedback (proactive rather than retroactive), i.e. they are not restricted to being applied only *a posteriori* (Potthast et al., 2017) and (2) they are scalable. A guiding assumption of language-based approaches is that there are statistical regularities inherent in natural languages and distributional patterns of language use indicative of fake news that are not consciously accessible to fake news creators. Space limitations prevent us from going into further details (but see reviews and survey cited above). In what follows, we will zoom in on previous studies on fake news detection conducted on the bases of the publicly available benchmark datasets used in the corpus study: the ISOT dataset, an 'entire article' dataset comprising 20k+ real and fake news texts (Ahmed et al., 2018), and the LIAR dataset, a 'claims dataset' comprising 12k+ real-world short statements collected from a variety of online sources (Wang, 2017) (see section 3 for details). Upon introduction of the ISOT dataset, (Ahmed et al., 2018) report on the results of experiments using n-gram features with two different features extraction techniques - Term Frequency (TF) and Term Frequency-Inverted Document Frequency (TF-IDF) - and six different machine learning techniques - Stochastic Gradient Descent, Support Vector Machines, Linear Support Vector Machines (LSVM), K-Nearest Neighbour and Decision Trees. Their best-performing model reached a classification accuracy of 92% using TF-IDF for feature extraction and an LSVM classifier, showing that real and fake news can be discriminated with high accuracy on the basis of the use of multiword sequences. However, subsequent studies have demonstrated that classification accuracy on this dataset can be pushed even higher - beyond the 99% accuracy mark - through the employment of deep neural networks trained on word embedding vectors: (Kula et al., 2020) reported classification accuracy between 95.04% and 99.86% using an LSTM neural network trained on different word em-

beddings (glove, news, Twitter, crawl) implemented in the Flair NLP framework (Akbik et al., 2019). Goldani et al., (2020) achieved a classification accuracy of 99.8% using a non-static capsule network and 'glove.6B.300d' word embeddings (Pennington et al., 2014). While the dataset ISOT involves a binary classification (fake vs. real), the LIAR dataset presents a six-way multiclass classification problem, where individual claims statement was evaluated for its truthfulness and received a much more fine-grained veracity label. In the experiments presented upon publication of the LIAR dataset, (Wang, 2017) provided several benchmarks based on several shallow learning classifiers (e.g. logistic regression and support vector machines) trained on n-gram features and deep learning classifiers (bi-directional long short-term memory and convolutional neural networks architectures) using pre-trained 300-dimensional word2vec embeddings from Google News (Mikolov et al., 2013). The latter reached a classification accuracy task of up to 27%. Incorporating available meta-data about the subject, speaker and context raised classification accuracy to 27.4%. Subsequent studies have shown that the classification accuracy on the LIAR set can be further increased to just over 45% by more complex hybrid models that integrate the linguistic information with speaker profiles into an attention based LSTM model (Long, 2017), by supplementing the data with verdict reports written by annotators (Karimi et al., 2018) or by replacing the credibility history in LIAR with a larger credibility source (Kirilin and Strube, 2018). Importantly, however, all state-of-the-art models designed to detect the veracity of a news article or claim exploit the information contained in high-dimensional word embeddings that are uninterpretable to humans, thereby severely limiting our ability to understand 'why' a given claim or news article was predicted to be fake or real.

3 Data

The experiments were conducted on two recently released datasets for fake news detection, the ISOT dataset compiled by the Information Security and Object Technology research lab (Ahmed et al., 2018) and the LIAR dataset introduced in (Wang, 2017). The datasets were selected based on their complementary attributes in terms of text types (full articles with average length of about 400 words vs. short statements with an average length of just under 20 words) and the granularity of the veracity labels (binary labels based on source selection and six-way classification based on ratings by politifac.com editors). Both datasets are sufficiently large for training deep models. The ISOT dataset consists of 40,000+ real and fake news articles collected from real-world sources between 2016 and 2017. The real (truthful) news articles were obtained by crawling articles from Reuters.com. The fake news articles were collected from unreliable websites that were flagged by politifact.com, a fact-checking organization in the USA, and Wikipedia. The ISOT dataset contains articles on a variety of topics with a focus on political and world news topics (see Table 2). For each article the following information is provided: article title, text, type (topic) and publication date. Close inspection of the dataset revealed that all and only instances of real news were introduced by the words "WASHINGTON (Reuters)", indicating the place and name of the news agency that has provided the news article. To prevent our models from capitalizing on this information, all instances of this string were deleted. We also checked for and removed all duplicates in the dataset (N = 6251). Table 2 presents the distribution of articles across news types (real/fake) and topics before and after deduplication (original/cleaned). The dataset was split in training, development, and testing sets using a 80/10/10 split. The LIAR dataset is a recent benchmark dataset for fake news detection that in includes 12,836 real-world short statements collected from a variety of online sources - including Facebook posts, tweets, news releases, TV/radio interviews, campaign speeches, TV ads and debates - on a range of topics - including economy, healthcare, taxes, federal-budget, education, jobs, state budget, candidates-biography, elections, and immigration. Each statement was labeled by an editor from politifact.com on a six-level ordinal scale of truthfulness ranging from "True", for completely accurate statements, to "Pants on Fire" (from the taunt "Liar, liar, pants on fire") for false and ludicrous claims. The distribution of the six labels is relatively well-balanced: with the exception of 1,050 instances of the 'pants-fire' category, the instances for all other labels range from 2,063 to 2,638. The LIAR set further includes a rich set of meta-data for each speaker including party affiliation, current job and home state. The statements in the dataset are also fairly balanced across the two major political par-

ties of the US - democrats and republicans - and also contain a significant amount of posts from online social media. The dataset is distributed into training, validation and testing sets in a 80/10/10 manner.

4 Automated Text Analysis

The raw texts from the two datasets were automatically analyzed using CoCoGen, a computational tool that implements a sliding window technique to calculate within-text distributions of feature scores (see recently published papers that use this tool, (Ströbel et al., 2018; Kerz et al., 2020b; Kerz et al., 2020a). In contrast to the standard approach implemented in other tools for automated text analysis that rely on aggregate scores representing the average value of a feature in a text, the sliding-window approach generates a series of measurements representing the 'local' distributions of scores. A sliding window can be conceived of as a window of size ws, which is defined by the number of sentences it contains. The window is moved across a text sentence-by-sentence, computing one value per window for a given feature. The series of measurements faithfully captures a typically non-uniform distribution of features within a text and is referred here to as a 'contour'.[1] To compute the value of a given feature in a given window m ($w(m)$), a measurement function is called for each sentence in the window and returns a fraction (wn_m/wd_m). CoCoGen uses the Stanford CoreNLP suite (Manning et al., 2014) for performing tokenization, sentence splitting, part-of-speech tagging, lemmatization and syntactic parsing (Probabilistic Context Free Grammar Parser (Klein and Manning, 2003)). In its current version, CoCoGen supports a total 154 of features that fall into six categories: (1) features of syntactic complexity (N=19), (2) features of lexical density, sophistication and variation (N=12), (3) information-theoretic features (N=3), (4) register-based n-gram frequency features (N=25), (5) LIWC-style (Linguistic Inquiry and Word Count) features (N=61) and (6) Word-Prevalence measures (N=36). A brief overview of the features and their short descriptions are provided in Table 4 in the Appendix. The inclusion of these features[2] is motivated by contemporary language and cognitive sciences characterized by an integrated, multi-method, and transdisciplinary approach needed to advance our understanding of the human processing and learning mechanisms (Christiansen and Chater, 2017). The first three sets of features are derived from the literature on language development showing that, in the course of their lifespan, humans learn to produce and understand complex syntactic structures, more sophisticated and diverse vocabulary and informationally denser language (see, e.g., Berman, 2007; Lu, 2010, 2012; Hartshorne and Germine, 2015; Ehret and Szmrecsanyi, 2019). The fourth set of features is derived from research on language adaptation (Chang et al., 2012) and research that looks at language from the perspective of complex adaptive systems (Beckner et al., 2009; Christiansen and Chater, 2016) indicating that, based on accumulated language knowledge emerging from lifelong exposure to various types of language inputs, humans learn to adapt their language to meet the functional requirements of different communicative contexts. The features in set five are based on insights from many years of research conducted by Pennebaker and colleagues (Pennebaker et al., 2003; Tausczik and Pennebaker, 2010), showing that the words people use in their everyday life provide important psychological cues to their thought processes, emotional states, intentions, and motivations. And finally, the inclusion of features in group six is motivated by recent efforts to estimate of of how well words are known in the population through crowdsourcing and corpus-based techniques. An accumulating body of evidence shows that such word prevalence measures are good predictors of human perfomance on various language tasks (Brysbaert et al., 2019; Johns et al., 2020)

5 Classification Models

For the classification, we used Bi-directional Recurrent Neural Network (BRNN) classifiers with Gated Recurrent Unit (GRU) cells (Cho et al., 2014). BRNNs have been shown to outperform unidirectional RNNs in application areas ranging from acoustic modeling (Sak et al., 2014) to machine translation (Bahdanau et al., 2014). Bi-directional neural network models have also been employed in previous

[1] In general, for a text comprising n sentences, there are $w = n - ws + 1$ windows. Given the constraint that there has to be at least one window, a text has to comprise at least as many sentences at the ws is wide $n \geq w$.

[2] CoCoGen was designed with extensibility in mind, so that additional features can easily be implemented. It uses an abstract measure class for the implementation of additional features.

studies on the two datasets investigated here (Wang, 2017; Kula et al., 2020), making them well suited for purposes of comparison and, more specifically, for examining whether and to what extent a classifier trained on human-interpretable features can approximate the performance of a state-of-the-art classifier trained on word embeddings. Since the two datasets differ in terms of the availability of meta-data (ISOT: no meata-data, LIAR: rich information on subject, speaker and context) and with respect to the granularity at which truthfullness was assessed (ISOT: binary, LIAR: 6-way multiclass), the BRNN classifiers were adapted so as to take these differences into account.Figure 1 shows the architecture of models used in the present paper. $X = (x_1, x_2, \ldots, x_n)$ is the output from CoCoGen, which is a sequence of 154-dimensional vectors. To integrate the context information, the words in the context description were mapped to 300-dimensional word embedding vectors using the dependency based word-embedding implemented in spaCy (Honnibal and Montani, 2017), represented by $C = (c_1, c_2, \ldots, c_n)$. Instead of one-hot encoding, we use word embeddings and BRNN to encode the context meta information here, as otherwise the feature vector for context information would result in 5075-dimensional sparse one-hot vectors. $J = (j_1, j_2, \ldots, j_n)$ is a sequence of word embeddings for the job title of the speaker of a given text, following the same reasoning as above. $S = (s_1, s_2, \ldots, j_n)$ and $P = (p_1, s_2, \ldots, p_n)$ are 70 and 25 dimensional on-hot vector for state information and party affiliation of the speaker. The structure of the classifier for ISOT dataset is shown in 1 on the left hand side in Figure 1. The lower part encircled by the dashed red line represents the recurrent network, where the CoCoGen output for a given text is fed into a 2-layer BRNN consisting of GRU cells with 200 hidden units in each layer. h_{10}, h_{20} represent the initial hidden states of the first and second layer of the BRNN respectively in the forward direction and h'_{10}, h'_{20} represent the initial hidden states of the first and second layer of BRNN respectively in the backward direction. h_{2n} and h'_{2n} represent the last hidden states of the second layer of the BRNN in the forward and backward direction respectively. These layers are concatenated and passed through a feed-forward neural network, encircled by the blue dashed line in Figure 1. This network consists of three linear layers, whose output dimensions are 200, 100 and 2. Between layers 1 and 2 as well as between layers 2 and 3 we inserted a Batch Normalization (BN) layer, a Parametric ReLU (PReLU) activation function layer and a Dropout layer with a dropout rate of 0.5. A softmax layer is applied before the final output $\hat{y}$. For the LIAR dataset, we built three BRNN models: (1) a model using only the CoCoGen output (X), (2) a model using CoCoGen output and the context information (X + C) and (3) a model using CoCoGen output, the context information and the speaker profile, which comprises information about the job, the state and the party of the of a speaker (X + C + J + S + P). The structure of CoCoGen-only model is identical to model built for the ISOT dataset, with the exception that the output layer has a size of 6 instead of 2. In the CoCoGen + Context model shown in sub-figure 2 in Figure 1, the sequence vector $X = (x_1, x_2, \ldots, x_n)$ represents the CoCoGen output as described for the ISOT model above. BRNN blocks in sub-figure 2 has a same structure as the lower part of sub-figure 1, which is a 2-layer bidirectional RNN, whose output is a concatenation of the last hidden state of uppermost layer in forward and backward direction respectively. The BRNN on the left size in sub-figure 2 has a hidden state size of 200, while the BRNN on the right side has one of 10. The Feed-forward 1 block is identical to the Feed-forward part shown in sub-figure 1. Sub-figure 3 shows the structure of model making use of CoCoGen features + context + speaker profiles. S and P are one-hot encoded vectors described as above. They are squeezed to 10-dimensional vectors through a feed-forward neural network, Feed-forward 2, whose structure is shown in the lower right part of Figure 1. Feed-forward 2 consists of two linear layers, the output of which are 20 for Linear 1 and 10 for Linear 2 respectively. The BRNN for CoCoGen output and context are identical to the corresponding BRNN blocks mentioned above. The BRNN for job title information encoding has the same structure and hidden state size as BRNN for context. All output from BRNN blocks and Feed-forward 2 blocks are concatenated and fed into Feed-forward 1 block, whose structure is shown in the upper part of sub-figure **1** with the exception that linear layers have output size of 210, 105 and 5 respectively. Since the labels of the LIAR dataset are ordinal in nature, i.e. pants-fire $<$ false $<$ barely-true $<$ half-true $<$ mostly-true $<$ true, the classification of instance in liar dataset can be treated as an ordinal classification problem. To adapt the neural network classifier to the ordinal classification task, we followed the NNRank approach described in (Cheng et al., 2008), which

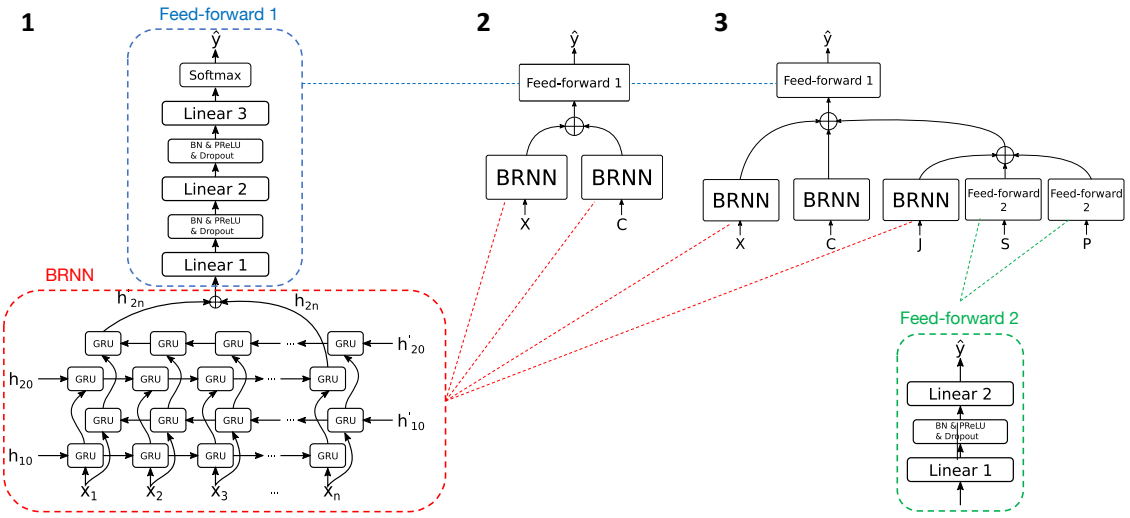

Figure 1: Structure of the BRNN classifiers built for the ISOT and LIAR datasets: The structure in 1 represents the model architecture used for ISOT and LIAR that makes use of textual information only (all CoCoGen features). The structures in 2 and 3 represent the model extensions that incorporate contextual meta-data (C) and speaker profiles (J = job title, P = party affiliation, S = speaker).

is a generalization of ordinal perception learning in neural networks(Crammer and Singer, 2002) and outperforms a neural network classifier on several benchmark datasets. Instead of one-hot encoding of class labels and using softmax as the output layer of a neural network, in NNRank, a class label for class k is encoded as $(y_1, y_2, \ldots, y_i, \ldots, y_{C-1})$, in which $y_i = 1$ for $i \leq k$ and $y_i = 0$ otherwise, where C is the number of classes. For the output layer, a sigmoid function was used. For prediction, the output of the neural network $(o_1, o_2, \ldots, o_{C-1})$ is scanned from left to right. It stops after encountering o_i, which is the first element of the output vector that is smaller than a threshold T (e.g. 0.5), or when there is no element left to be scanned. The predicted class of the output vector is the index k of the last element, whose value is greater than or equal to T. Finally, for the purpose of comparison, we also recreated the convolutional neural network (CNN) model described in (Wang, 2017). This CNN model consists of filters of size 2, 3 and 4. Each size has 128 filters with a max-pooling operation being performed on each output filter. The result of the max-pooling was fed into a feed-forward neural network for the classification. As an additional baseline, we further built structurally equivalent BRNN classifiers based on sentence embeddings from Sentence-BERT (SBERT) (Reimers and Gurevych, 2019).[3]

All models are implemented using PyTorch (Pytorch, 2019). For the BRNNs and the CNN that don't use the ordinal information cross entropy loss was used as a loss function:

$$\mathcal{L}(\hat{Y}, c) = -\sum_{i=1}^{C} p(y_i) \log(p(\hat{y}_i))$$

where c is the true class label of the current observation, C is the number of classes, $(p(y_1), \ldots, p(y_C))$ is a one-hot vector with

$$p(y_i) = \begin{cases} 1 & i = c \\ 0 & \text{otherwise} \end{cases}$$

and $\hat{Y} = (p(\hat{y}_1), p(\hat{y}_2), \ldots, p(\hat{y}_C))$ is the output vector of the softmax layer, which can be viewed as the predicted probabilities of the observed instance falling into to each of the classes. For training BRNNs

[3]SBERT is a finetuned BERT network using siamese and triplet network structures that. It has been shown to outperform other state-of-the-art sentence embeddings methods on common semantic textual similarity and transfer learning tasks (Reimers and Gurevych, 2019).

using ordinal information binary cross entropy was used:

$$\mathcal{L}(\hat{Y}, c) = -\frac{1}{C}\sum_{i=1}^{C}(y_i \log(\hat{y}) + (1 - y_i)\log(1 - \hat{y}))$$

in which $c = (y_1, y_2, \ldots, y_N), C = 14$ is number of responses and $\hat{Y} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_N)$ is the output vector of the sigmoid layer rounded to closest integer. We tuned all hyperparameters on the validation set using a grid search over sets of optimizers $S = \{\text{Adamax, Adagrad, RMSprop}\}$, learning rates $L = \{0.01, 0.001, 0.0001\}$ and normalization methods $N = \{\text{Standardization, Min-max}\}$. The optimal hyperparameter combinations are provided in Table 5 in the Appendix.

To determine the relative importance of the language features groups, we conducted feature ablation experiments. Classical forward or backward sequential selection algorithms that proceed by sequentially adding or discarding features require a quadratic number of model training and evaluation in order to obtain a feature ranking (Langley, 1994). In the context of neural network models, training a quadratic number of models can become prohibitive. To alleviate this problem, we used an adapted version of the iterative sensitivity-based pruning algorithm proposed by (Díaz-Villanueva et al., 2010). This algorithm ranks the features based on a 'sensitivity measure' (Moody, 1994; Utans and Moody, 1991) and removes the least relevant variables one at a time. The classifier is then retrained on the resulting subset and a new ranking is calculated over the remaining features. This process is repeated until all features are removed. In this fashion, rather than training $\frac{n(n+1)}{2}$ models required for sequential algorithms, the number of models trained is reduced to $\frac{n}{m}$, where m is the number of features or feature groups that can be removed at each step. We report the results obtained after the removal of a single feature group at each step. At step t, a neural network model M_t is trained on the training set. The training set at step t consists of instances with feature groups $F_t = \{f_1, f_2, \ldots, f_{D_t}\}$ where $f_1, \ldots f_{D_t}$ are the remaining feature groups at the current step, whose importance rank is to be determined. We define X_t as the test set with feature set F_t and X_t^i as the same dataset as X_t except we set the i^{th} feature f_i of each instance within the dataset to its average. Furthermore, we define $g(X)$ as the classification accuracy of $M_{t,n}$ for a dataset X. The sensitivity of a feature group f_i at step t is obtained from:

$$S_{i,t} = g(X_t) - g(X_t^i)$$

The most important feature group at step t can be found by:

$$f_{\hat{i}} : \hat{i} =_{i:f_i \in F_t} (S_{i,t})$$

Then we set the rank for feature $f_{\hat{i}}$:

$$Rank_{\hat{i}} = t$$

In the end, feature $f_{\hat{i}}$ is dropped from F_t and the corresponding columns in training and test dataset are also dropped simultaneously:

$$F_{t+1} = F_t - \{f_{\hat{i}}\}$$

This procedure is repeated, until $|F_{t'}| = 1$.

6 Results

The performance metrics of the classification models for both datasets (global accuracy, precision and recall) are presented in Table 1, along with comparisons with the results of previous studies (a extended version of the table with performance data of additional models is provided in the Appendix). The results of our BRNN classifiers trained on interpretable features are highly competitive with those obtained from state-of-the-art RNN, CNN and capsule networks that exploit word embeddings to represent textual contents. In fact, in both datasets, our classifiers match the performance of the best-performing models within half a percent: For ISOT, the CAPSULE-glove (Goldani et al., 2020) and LSTM-glove (Kula et al., 2020) both achieve an accuracy of 99.8%, while BRNN CoCoGen achieves 99.3%. Moreover, the BRNN CoCoGen model outperformed the LSTMs presented in Kula et al. (2020) that utilize three other

word embeddings implemented in the Flair library (news, Twitter, crawl) by up to 4.3% and improved on the performance on the n-gram-based LSVM model by 7.3%. For the LIAR data set, the difference in classification accuracy between BRNN CoCoGen and the CNN utilizing 300-dimensional word embeddings trained on Google News presented in Wang (2017) amounts to 0.2%, when meta-data on context and speaker profiles is taken into account. Excluding all meta-data, the BRNN CoCoGen (ordered) model reached an accuracy of 27.7%, which is even slightly higher than the performance of the Bi-LSTM 300-dim word2vec embeddings (Google news) model. Our CNN CoCoGen model achieved a classification accuracy of 25.6%, which is 1.4% below the performance of the corresponding CNN model presented in Wang (2017) , CNN 300-dim word2vec embeddings (Google News). Interestingly, however, this model suffered from a substantial drop in accuracy to 24.8%, once it was infused with contextual meta-data. In contrast, all BRNN CoCoGen models invariably benefited from the addition of any type of meta-data. While performance with the CAPSULE-glove networks presented in Goldani et al. (2020) is limited by their selective integration of meta-data, it is worth noting that the CNN CoCoGen model outperformed all their models without recourse to meta-data. Taken together these results present strong evidence that successful detection of fake news can be achieved without sacrificing transparency. It is also worth pointing out that approaching the fake news detection task as an ordinal classification problem had considerable effects on a classifiers performance. Specifically, we observed (1) that classification accuracy slightly increased by 0.6% relative to a unordered classification approach and (2) that classification behavior shifted from a bias towards recall to a bias towards precision. Furthermore, comparison of the confusion matrices of our classifiers revealed that changing to the ordinal classification approach had positive effects on the distribution of errors: The ordinal classification problem is monotonic, meaning that the further a misclassification is from the main diagonal of a confusion matrix, the more severe it is. The confusion matrix of the best-performing BRNN CoCoGen model shows that for five out of the six classes (pants-fire, false, half-true, mostly-true, true) the most frequent prediction was the true class and the number of misclassifications decreases with increasing distance to the true class. In contrast, in the case of the unordered classifiers, we observed that the extreme categories ('pants-fire' and 'true') were avoided and predictions to the intermediate categories were preferred, especially in classifiers without meta-data information (confusion matrices for all models are provided in the Appendix). To the best of the authors knowledge, current models on multi-class fake news detection do not concern with the order of labels (Oshikawa et al., 2018). Our results indicate that future work can benefit from taking an ordinal classification approach. The results of our feature ablation experiments revealed a similar rank order in feature importance in both datasets (detailed results can be found in Table 14 in the Appendix): In each case, classification performance was mainly driven by features from the groups Lexical, LIWC, Syntactic and register-based n-grams, and to a lesser extent by information theoretic and word-prevalence-based features. Specifically, Table 14 indicates that - in the casse of the ISOT dataset - dropping the features from the LIWC group results in the largest decrease in classification accuracy of 5.1% on the validation set, resulting in a drop in accuracy on the test set to 93.8%. Re-training the model without the LIWC features yields the new baseline of 99.1%, indicating that the remaining features contained enough information to allow the retrained model to compensate for the loss of the LIWC information. After the elimination of the next two most-important feature groups, the syntactic and lexical groups, the retrained model at iteration 3 is still able to achieve an accuracy on the validation set of 97.9%. However, after the drop of the n-gram feature group, classification accuracy on the drops to 76.3% (validation) and 76.2% (test), indicating that the lost information from the four top-feature groups cannot be compensated for by information from the remaining feature groups, i.e. information theoretic and word-prevalence-based features. In the case of the the LIAR dataset, the relative influence of the six feature groups is more even and the predictive power of the model (27.2% accuracy on the test set) appears to stem from exploiting information from all six feature groups. For a closer examination of how individual features within each feature-group distinguished between real and fake news, we derived standard scores by performing z-standardization on all indicators and determined the difference between mean standard scores of real and fake news ($DeltaScore_{index\ i}$ = $Score_{index\ i,\ fake\ news} - Score_{index\ i,\ real\ news}$) (a complete table with the DeltaScores for the top-20 features for both datasets is provided in the Appendix). Inspection

		Validation set			Test set		
Dataset	Model	Accuracy	Precision	Recall	Accuracy	Precison	Recall
ISOT	LSVM unigram 50k[1]	–	–	–	0.920	–	–
	LSTM-glove[2]	–	–	–	**0.998**	–	–
	CAPSULE-glove[3]	–	–	–	**0.998**	–	–
	BRNN SBERT	0.998	0.998	0.998	0.997	0.997	0.997
	BRNN CoCoGen	0.994	0.994	0.994	**0.993**	0.993	0.993
LIAR	Bi-LSTM 300-dim word2vec[4] embeddings (Google News)	0.223	–	–	0.233	–	–
	CNN 300-dim word2vec[4] embeddings (Google News) + context + speaker profile	0.247	–	–	**0.274**	–	–
	CAPSULE-glove + Party[3]	0.261	–	–	0.240	–	–
	CAPSULE-glove + State[3]	0.240	–	–	0.243	–	–
	CAPSULE-glove + Job[3]	0.254	–	–	0.251	–	–
	BRNN SBERT (ordered)	0.292	0.272	0.327	0.270	0.296	0.249
	BRNN CoCoGen (ordered)	0.251	0.281	0.218	0.237	0.217	0.207
	BRNN CoCoGen (ordered) + context	0.264	0.280	0.241	0.253	0.281	0.238
	BRNN CoCoGen (ordered) + context + speaker profile	0.284	0.305	0.263	**0.272**	0.304	0.258

Table 1: Evaluation results on the ISOT and LIAR datasets on the validation and test sets.[1] = Ahmed et al., 2018; [2] = Kula et al., 2020; [3] = Goldani et al., 2020; [4] = Wang, 2017

of the Delta Scores revealed some interesting patterns. For example, real news articles and claims are characterized by (1) relatively higher lexical diversity (as measured by type-token ratio features), (2) stronger reliance of multiword sequences from the news and academic register (measured by register-based n-gram frequency measures), (3) greater phrasal syntactic complexity (as measured, e.g., by the number of complex nominals per clause) and (4) more frequent use or word from particular domains, such as work, money, power or word classes, such as preposition and quantifiers. In contrast, fake news are characterized by (1) greater syntactic complexity (as measured by, e.g. by the number of clauses per sentence), (2) frequent use of multiword sequences form the domain of fiction, (3) higher lexical sophistication scores (as measured in terms of relatively infrequent words) and (4) a strong reliance on personal pronouns, adverbs and emotion words. While limitations of space preclude an in-depth discussion, these results demonstrate that the use of interpretable features can provide new insights and knowledge about the characteristics of fake news and explain "why" a piece of news was detected as fake news see (Shu et al., 2019) for a discussion of explainable fake news detection.

7 Conclusion and Future Work

In recent years, there is a growing recognition of the need to move away from black-box models towards white-box models for solving practical problems, in particular in the context of critical industries, including healthcare, criminal justice, and news (Rudin, 2019). This is due to the fact that human experts in a given application domain need both accurate but also understandable models (Loyola-Gonzalez, 2019). In this paper, we have made a contribution to this development in the domain of fake news detection. We have demonstrated that models trained on human interpretable features in combination with deep learning classifiers can compete with black box models based on word embeddings. In the future we intend to extend this work in two directions: First, we plan to apply our approach to fake news detection in German whose research still lags far behind that available for English. Second, we also plan to apply our approach to the detection of rumours and conspiracy theories to tackle and combat the ongoing Covid-19 infodemic.

References

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

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–36.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Clay Beckner, Richard Blythe, Joan Bybee, Morten H Christiansen, William Croft, Nick C Ellis, John Holland, Jinyun Ke, Diane Larsen-Freeman, et al. 2009. Language is a complex adaptive system: Position paper. *Language learning*, 59:1–26.

Ruth A Berman. 2007. Developing linguistic knowledge and language use across adolescence.

Marc Brysbaert, Paweł Mandera, Samantha F McCormick, and Emmanuel Keuleers. 2019. Word prevalence norms for 62,000 english lemmas. *Behavior research methods*, 51(2):467–479.

Franklin Chang, Marius Janciauskas, and Hartmut Fitz. 2012. Language adaptation and learning: Getting explicit about implicit learning. *Language and Linguistics Compass*, 6(5):259–278.

Jianlin Cheng, Zheng Wang, and Gianluca Pollastri. 2008. A neural network approach to ordinal regression. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 1279–1284. IEEE.

KyungHyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *CoRR*, abs/1409.1259.

Morten H Christiansen and Nick Chater. 2016. *Creating language: Integrating evolution, acquisition, and processing*. MIT Press.

Morten H Christiansen and Nick Chater. 2017. Towards an integrated science of language. *Nature Human Behaviour*, 1(8):1–3.

Koby Crammer and Yoram Singer. 2002. Pranking with ranking. In *Advances in neural information processing systems*, pages 641–647.

Wladimiro Díaz-Villanueva, Francesc J Ferri, and Vicente Cerverón. 2010. Learning improved feature rankings through decremental input pruning for support vector based drug activity prediction. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 653–661. Springer.

Katharina Ehret and Benedikt Szmrecsanyi. 2019. Compressing learner language: An information-theoretic measure of complexity in sla production data. *Second Language Research*, 35(1):23–45.

Mohammad Hadi Goldani, Saeedeh Momtazi, and Reza Safabakhsh. 2020. Detecting fake news with capsule neural networks. *arXiv preprint arXiv:2002.01030*.

Joshua K Hartshorne and Laura T Germine. 2015. When does cognitive functioning peak? the asynchronous rise and fall of different cognitive abilities across the life span. *Psychological science*, 26(4):433–443.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Brendan T Johns, Melody Dye, and Michael N Jones. 2020. Estimating the prevalence and diversity of words in written language. *Quarterly Journal of Experimental Psychology*, 73(6):841–855.

Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. 2018. Multi-source multi-class fake news detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1546–1557.

Elma Kerz, Fabio Pruneri, Daniel Wiechmann, Yu Qiao, and Marcus Ströbel. 2020a. Understanding the dynamics of second language writing through keystroke logging and complexity contours. In *Proceedings of The 12th Language Resources and Evaluation Conference (LREC2020))*, pages 182–188.

Elma Kerz, Yu Qiao, Daniel Wiechmann, and Marcus Ströbel. 2020b. Becoming linguistically mature: Modeling english and german children's writing development across school grades. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications (BEA2020))*, pages 65–74.

Angelika Kirilin and Micheal Strube. 2018. Exploiting a speaker's credibility to detect fake news. In *Proceedings of Data Science, Journalism & Media workshop at KDD (DSJM'18)*.

Dan Klein and Christopher D Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st annual meeting of the association for computational linguistics*, pages 423–430.

Sebastian Kula, Michał Choraś, Rafał Kozik, Paweł Ksieniewicz, and Michał Woźniak. 2020. Sentiment analysis for fake news detection by means of neural networks. In *International Conference on Computational Science*, pages 653–666. Springer.

Pat Langley. 1994. Selection of relevant features in machine learning. In *Proceedings of the AAAI Fall symposium on relevance*, pages 1–5.

Yunfei Long. 2017. Fake news detection through multi-perspective speaker profiles. Association for Computational Linguistics.

Octavio Loyola-Gonzalez. 2019. Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view. *IEEE Access*, 7:154096–154113.

Xiaofei Lu. 2010. Automatic analysis of syntactic complexity in second language writing. *International Journal of Corpus Linguistics*, 15(4):474–496.

Xiaofei Lu. 2012. The relationship of lexical richness to the quality of ESL learners' oral narratives. *The Modern Language Journal*, 96(2):190–208.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Prismatic Inc, Steven J. Bethard, and David Mcclosky. 2014. The stanford corenlp natural language processing toolkit. In *In ACL, System Demonstrations*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

John Moody. 1994. Prediction risk and architecture selection for neural networks. In *From statistics to neural networks*, pages 147–165. Springer.

Ray Oshikawa, Jing Qian, and William Yang Wang. 2018. A survey on natural language processing for fake news detection. *arXiv preprint arXiv:1811.00770*.

James W Pennebaker, Matthias R Mehl, and Kate G Niederhoffer. 2003. Psychological aspects of natural language use: Our words, our selves. *Annual review of psychology*, 54(1):547–577.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2017. A stylometric inquiry into hyperpartisan and fake news. *arXiv preprint arXiv:1702.05638*.

Pytorch. 2019. Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration. `https://github.com/pytorch/pytorch`.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Hasim Sak, Andrew W Senior, and Françoise Beaufays. 2014. Long short-term memory recurrent neural network architectures for large scale acoustic modeling.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1):22–36.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 395–405.

Marcus Ströbel, Elma Kerz, Daniel Wiechmann, and Yu Qiao. 2018. Text genre classification based on linguistic complexity contours using a recurrent neural network. In *Proceedings of the Tenth International Workshop 'Modelling and Reasoning in Context' co-located with the 27th International Joint Conference on Artificial Intelligence (IJCAI 2018) and the 23rd European Conference on Artificial Intelligence*, pages 56–63.

Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54.

Joshua A Tucker, Andrew Guess, Pablo Barberá, Cristian Vaccari, Alexandra Siegel, Sergey Sanovich, Denis Stukal, and Brendan Nyhan. 2018. Social media, political polarization, and political disinformation: A review of the scientific literature. *Political polarization, and political disinformation: a review of the scientific literature (March 19, 2018)*.

Joachim Utans and John Moody. 1991. Selecting neural network architectures via the prediction risk: Application to corporate bond rating prediction. In *Proceedings First International Conference on Artificial Intelligence Applications on Wall Street*, pages 35–41. IEEE.

William Yang Wang. 2017. " liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*.

Ken Ward. 2018. Social networks, the 2016 us presidential election, and kantian ethics: applying the categorical imperative to cambridge analytica's behavioral microtargeting. *Journal of media ethics*, 33(3):133–148.

Xichen Zhang and Ali A Ghorbani. 2020. An overview of online fake news: Characterization, detection, and discussion. *Information Processing & Management*, 57(2):102025.

Xinyi Zhou, Reza Zafarani, Kai Shu, and Huan Liu. 2019. Fake news: Fundamental theories, detection strategies and challenges. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 836–837.

Appendix

Table 2: Composition of the ISOT dataset; sizes indicate the number of articles in a given category; 'cleaned' refers to the datasets after deduplication

News Type	Total size	Topic	Size
Real News	original: 21417; cleaned: 21192	World-News	original: 10145; cleaned: 9978
		Politics-News	original: 11272; cleaned: 11214
Fake-News	original: 23481: cleaned: 17455	Government-News	original: 1570; cleaned: 514
		Middle-east	original: 778; cleaned: 0
		US News	original: 783; cleaned: 783
		Left-News	original: 4459; cleaned: 683
		Politics	original: 6841; cleaned: 6425
		News	original: 9050; cleaned: 9050

Table 3: Composition of the LIAR dataset

Training set size	10,269
Validation set size	1,284
Testing set size	1,283
Avg. statement length (tokens)	17.9
Top-3 Speaker Aliations	
Democrats	4,150
Republicans	5,687
None (e.g., FB posts)	2,185

Table 4: Concise overview of the six feature groups

Feature group	Size	Subtypes	Example/Description
Syntactic complexity	18	Length of production unit Subordination Coordination Particular structures	e.g. mean length of clause e.g. clauses per sentences e.g. Coordinate phrases per clause e.g. Complex nominals per clause
Lexical richness	12	Lexical density Lexical diversity Lexical sophistication	e.g. ration contents words / all words e.g. type token ratio e.g. words on General Service List
Register-based n-gram frequency	25	Spoken ($n \in [1, 5]$) Fiction ($n \in [1, 5]$) Magazine ($n \in [1, 5]$) News ($n \in [1, 5]$) Academic ($n \in [1, 5]$)	measures of frequencies of n-grams of order 1-5 from five language registers
Information theory	3	$\text{Kolmogorov}_{\text{Deflate}}$ $\text{Kolmogorov}_{\text{Deflate Syntactic}}$ $\text{Kolmogorov}_{\text{Deflate Morphological}}$	measures use Deflate algorithm and relate size of compressed file to size of original file
LIWC-style	60	2300 words from > 70 classes	classes include e.g. function, grammar perceptual, cognitive and biological processes, personal concerns, affect, social, basic drives, ...
Word-Prevalence	36	crowdsourcing-based corpus-based	measures capture information on word frequency, contextual diversity and semantic distinctiveness differentiated across language variety (US, UK) and gender (male, female)

Dataset	Model	Optimizer	learning rate	Normalization Method
ISOT	BRNN	Adamax	0.001	Standardization
LIAR	BRNN (ordered)	Adamax	0.001	Standardization
	BRNN (unordered)	RMSprop	0.001	Min-Max
	CNN	RMSprop	0.01	Standardization
	BRNN + context	Adamax	0.0001	Standardization
	BRNN + context + speaker profile (unordered)	RMSprop	0.0001	Standardization
	BRNN + context + speaker profile (ordered)	Adamax	0.001	Standardization

Table 5: Optimal combinations of optimizer, learning rate and normalization methods identified via grid search.

		Validation set			Test set		
Dataset	Model	Accuracy	Precision	Recall	Accuracy	Precison	Recall
ISOT	LSVM unigram 50k (Ahmed et al. 2018)	–	–	–	0.920	–	–
	LSTM-glove	–	–	–	**0.998**	–	–
	LSTM-news	–	–	–	0.950	–	–
	LSTM-twitter	–	–	–	0.980	–	–
	LSTM-crawl (Kula et al., 2020)	–	–	–	0.976	–	–
	CAPSULE-glove (Goldani et al., 2020)	–	–	–	**0.998**	–	–
	BRNN SBERT (ordered)	0.292	0.272	0.327	0.270	0.296	0.249
	BRNN CoCoGen	0.994	0.994	0.994	**0.993**	0.993	0.993
LIAR	Bi-LSTM 300-dim word2vec embeddings (Google News)	0.223	–	–	0.233	–	–
	CNN 300-dim word2vec embeddings (Google News)	0.260	–	–	0.270	–	–
	CNN 300-dim word2vec embeddings (Google News) + context	0.277	–	–	0.248	–	–
	CNN 300-dim word2vec embeddings (Google News) + context + speaker profile (Wang, 2017)	0.247	–	–	**0.274**	–	–
	CAPSULE-glove + Party	0.261	–	–	0.240	–	–
	CAPSULE-glove + State	0.240	–	–	0.243	–	–
	CAPSULE-glove + Job (Goldani et al., 2020)	0.254	–	–	0.251	–	–
	BRNN SBERT (ordered)	0.292	0.272	0.327	0.270	0.296	0.249
	BRNN CoCoGen (unordered)	0.269	0.186	0.227	0.244	0.172	0.207
	BRNN CoCoGen (ordered)	0.251	0.281	0.218	0.237	0.217	0.207
	CNN CoCoGen (unordered)	0.266	0.357	0.224	0.256	0.155	0.216
	BRNN CoCoGen (ordered) + context	0.264	0.280	0.241	0.253	0.281	0.238
	BRNN CoCoGen (unordered) context + speaker profile	0.288	0.233	0.253	0.266	0.217	0.231
	BRNN CoCoGen (ordered) + context + speaker profile	0.284	0.305	0.263	**0.272**	0.304	0.258

Table 6: Evaluation results on the ISOT and LIAR datasets on the validation and test sets. Models indexed as "CoCoGen" comprise textual features only. Models with "+" are hybrid models with textual and meta-data. The labels "ordered" and "unordered" indicate whether an ordinal and nominal classification method was applied.

Table 7: Add caption

	Top-20 Measures Fake News			
	LIAR		**ISOT**	
	Measure	Delta Score	Measure	Delta Score
1	Lexical Density	0.183	LIWC Adverb	0.894
2	LIWC Focus future	0.132	LIWC Ipron	0.733
3	LIWC Relig	0.117	ngram 2 fic	0.728
4	Lexical Div CNDW	0.115	LIWC You	0.681
5	Lexical Div TTR	0.115	LIWC Focuspresent	0.677
6	Lexical Soph BNC	0.114	Mor Kolmogorov	0.670
7	LIWC Verb	0.106	Syntactic ClausesPerSentence	0.664
8	LIWC Hear	0.099	Base Kolmogorov	0.652
9	MeanLengthWord	0.098	LIWC Certain	0.630
10	Base Kolmogorov	0.095	Syntactic Kolmogorov	0.629
11	LIWC Negate	0.091	ngram 3 fic	0.620
12	Lexical Soph ANC	0.091	LIWC See	0.561
13	LIWC Posemo	0.087	Syntactic DepClausesPerTUnit	0.502
14	Morphological Kolmogorov	0.087	LIWC Interrog	0.498
15	Syntactic Kolmogorov	0.084	LIWC I	0.438
16	Syntactic VerbPhrasesPerTUnit	0.081	LIWC They	0.431
17	MeanSyllablesPerWord	0.076	LIWC Shehe	0.420
18	Lexical Soph NGSL	0.075	LIWC Female	0.385
19	LIWC Risk	0.062	LIWC Swear	0.380
20	LIWC Focuspresent	0.059	ngram 1 fic	0.370
	Top-20 Measures Real News			
	LIAR		**ISOT**	
	Measure	Delta Score	Measure	Delta Score
1	LIWC Quant	-0.212	Syntactic ComplexNomPerClause	-0.940
2	LIWC Compare	-0.197	Syntactic MeanLengthClause	-0.929
3	LIWC Adj	-0.171	LIWC Prep	-0.763
4	ngram 2 news	-0.148	LIWC Hear	-0.747
5	ngram 2 acad	-0.147	LIWC Power	-0.739
6	ngram 2 mag	-0.145	LIWC Work	-0.734
7	LIWC Time	-0.133	LIWC Article	-0.723
8	WordPrevalence	-0.132	LIWC Focus past	-0.666
9	ngram 1 acad	-0.131	LIWC Space	-0.545
10	ngram 3 acad	-0.128	Syntactic CoordPhrasesPerClause	-0.509
11	ngram 3 mag	-0.123	NP PreModWords	-0.493
12	ngram 3 news	-0.122	Lexical Div TTR	-0.465
13	ngram 1 mag	-0.120	Lexical Div CNDW	-0.465
14	ngram 1 fic	-0.118	Lexical Div RTTR	-0.413
15	ngram 1 news	-0.118	Lexical Div CTTR	-0.403
16	LIWC Number	-0.116	LIWC Money	-0.314
17	ngram 1 spok	-0.111	Lexical Soph BNC	-0.309
18	LIWC Space	-0.110	ngram 5 news	-0.306
19	LIWC Prep	-0.106	LIWC Achieve	-0.303
20	ngram 2 spok	-0.105	Lexical Density	-0.302

	pants-fire	false	barely-true	half-true	mostly-true	true
pants-fire	1	23	24	31	11	2
false	2	49	55	84	53	6
barely-true	3	36	62	80	27	4
half-true	1	38	58	108	55	5
mostly-true	1	19	41	104	74	2
true	3	19	43	71	66	6

Table 8: confusion matrix of liar dataset BRNN model

	pants-fire	false	barely-true	half-true	mostly-true	true
pants-fire	0	43	2	31	16	0
false	0	107	4	65	73	0
barely-true	0	78	8	61	65	0
half-true	0	83	9	84	89	0
mostly-true	0	44	3	84	110	0
true	0	60	0	60	88	0

Table 9: confusion matrix of liar dataset BRNN model (non-ordinal)

	pants-fire	false	barely-true	half-true	mostly-true	true
pants-fire	0	42	1	29	20	0
false	0	114	3	57	75	0
barely-true	0	86	2	71	52	1
half-true	0	88	4	97	76	0
mostly-true	0	77	2	51	111	0
true	0	74	1	46	87	0

Table 10: confusion matrix of liar dataset CNN model

	pants-fire	false	barely-true	half-true	mostly-true	true
pants-fire	15	29	22	14	11	1
false	8	57	65	62	49	8
barely-true	8	38	62	66	33	5
half-true	5	35	72	99	50	4
mostly-true	3	22	50	86	77	3
true	3	18	51	68	58	10

Table 11: confusion matrix of liar dataset BRNN model meta context

	pants-fire	false	barely-true	half-true	mostly-true	true
pants-fire	20	27	18	14	12	1
false	8	54	64	74	44	5
barely-true	5	39	59	65	42	2
half-true	2	32	46	111	69	5
mostly-true	1	21	36	85	94	4
true	3	25	31	64	79	6

Table 12: confusion matrix of liar dataset BRNN model all meta

	pants-fire	false	barely-true	half-true	mostly-true	true
pants-fire	0	49	17	9	14	3
false	0	108	20	54	38	29
barely-true	0	73	27	55	42	15
half-true	0	76	23	83	61	22
mostly-true	0	43	15	70	88	25
true	0	57	12	45	63	31

Table 13: confusion matrix of liar dataset BRNN model all meta (non-ordinal)

Dataset	Feature Group	Accuracy base model (validation)	Accuracy after drop (validation)	Accuracy after drop (test)
ISOT	LIWC	0.993	0.942	0.938
	Syntactic	0.991	0.964	0.965
	Lexical	0.988	0.915	0.912
	N-grams	0.989	0.763	0.762
	Info theory	0.979	0.822	0.823
	Word-prevalence	0.933	0.482	0.475
LIAR	Lexical	0.255	0.217	0.209
	LIWC	0.252	0.204	0.192
	Syntactic	0.232	0.193	0.215
	N-grams	0.224	0.188	0.218
	Word-prevalence	0.210	0.190	0.205
	Info theory	0.209	0.193	0.208

Table 14: Results of the feature ablation experiments for the ISOT dataset (top) and the LIAR dataset (bottom).