Expert Systems With Applications 128 (2019) 201–213

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Behind the cues: A benchmarking study for fake news detection

Georgios Gravanis[a,*], Athena Vakali[a], Konstantinos Diamantaras[b], Panagiotis Karadais[b]

[a] Aristotle University of Thessaloniki, University Campus, 54124, Thessaloniki, Greece
[b] A.T.E.I. of Thessaloniki P.O BOX 141, GR - 57400, Thessaloniki, Greece

ARTICLE INFO

Article history:
Received 24 October 2018
Revised 20 March 2019
Accepted 20 March 2019
Available online 21 March 2019

Keywords:
Fake news
Linguistic analysis
Text classification
Machine learning
Ensemble machine learning

ABSTRACT

Fake news has become a problem of great impact in our information driven society because of the continuous and intense fakesters content distribution. Information quality in news feeds is under questionable veracity calling for automated tools to detect fake news articles. Due to many faces of fakesters, creating such tool is a challenging problem. In this work, we propose a model for fake news detection using content based features and Machine Learning (ML) algorithms. To conclude in most accurate model we evaluate several feature sets proposed for deception detection and word embeddings as well. Moreover, we test the most popular ML classifiers and investigate the possible improvement reached under ensemble ML methods such as AdaBoost and Bagging. An extensive set of earlier data sources has been used for experimentation and evaluation of both feature sets and ML classifiers. Moreover, we introduce a new text corpus, the "UNBiased" (UNB) dataset, which integrates various news sources and fulfills several standards and rules to avoid biased results in classification task. Our experimental results show that the use of an enhanced linguistic feature set with word embeddings along with ensemble algorithms and Support Vector Machines (SVMs) is capable to classify fake news with high accuracy.

1. Introduction

Fake news spreading has lately become a crucial problem of large impact due to its unexpected consequences and its large-scale turmoil triggering. Historically, its roots go back to the 17th century in the form of "Propaganda" which was turned to the "Misinformation" of the Cold War era. Today's so-called "fake news" phenomenon refers to the online publication of intentionally or knowingly false statements of facts, and it has recently dominated current social media platforms (Klein & Wueller, 2017). Fake news are typically produced by people, the so-called "fakesters", who generate an article with fake content often injected to an original real and trusted news content. The term "fake news" was characterized as the most frequently used one in Social Media during 2016 and has gained popularity especially after the 2016 US election (Newman, 2017), while at a recent post by Collins Language Publications "fake news" was announced as the "word of the year" (this word's usage increased by 365% since 2016). This explosion in the search of the "fake news" term indicates its high daily and massive impact on people who are strongly engaged in Social Media. The rapid distribution of fake news is due to the widespread use of Social Media which offer a fertile ground for instantly sharing and circulating news with the users having no means of quality checking over the shared content. Since social media news updating has become the norm, it is evident that people are exposed to fake news content on a daily basis, news content quality becomes questionable and people's trust in news circulated universally is losing ground.

Fake news *detection* is a challenging problem and it is studied as a phenomenon with many angles. The complexity of the problem is due to the many faces of the fakesters who aim to deceive the reader for various unforeseeable reasons such as pecuniary or ideological etc. (Allcott & Gentzkow, 2017). This fact is emphasized by journalists who have spotted that fake news producers do not care about the damage done to politicians' profiles, but instead, their incentive is purely economic[1]. From another perspective, as recently highlighted by Poynter institute[2], it is often the case that satire articles or hoaxes can be transformed into fake news which flood the Web. Already, there are several examples of well-known Journalism organizations that reproduced such articles as real with

* Corresponding author.
E-mail addresses: ggravanis@csd.auth.gr (G. Gravanis), avakali@csd.auth.gr (A. Vakali), kdiamant@it.teithe.gr (K. Diamantaras), pkaradais@gmail.com (P. Karadais).

https://doi.org/10.1016/j.eswa.2019.03.036
0957-4174/

[1] https://www.buzzfeed.com/craigsilverman/how-macedonia-became-a-global-hub-for-pro-trump-misinfo.
[2] https://www.poynter.org/news/week-fact-checking-how-satirical-post-became-fake-news.

the most recent to be the "death of Costa Gavras"[3], a famous director who has been "reported" dead by a fake Twitter account.

Fake news spreading impacts many and important aspects of life, society, politics, and economy. Subsequently, the necessity of an automated tool classifying a piece of news according to its quality is evident. The first step towards drafting such a tool is to sketch the profile of fakesters who tend to use a variety of practices in order to achieve deception. Fakesters practices include:

(i) *hoax* i.e. plans to deceive, especially by playing a trick on someone.
(ii) *satire* i.e. ways of criticizing content, people or ideas in a humorous manner.
(iii) *fake news posting* i.e. the online publication of intentionally or knowingly false statements of fact.

Currently, the main stakeholders in the battle against fake news are fact checking organizations such as Snopes[4], Politifact[5], TruthOrFiction[6] etc. These organizations operate on the basis of the traditional journalistic model, in which reporters have to evaluate facts in order to obtain the veracity of a news snippet. This approach is not automated and is often time-consuming and difficult to compete with the quantity of fake news published daily. To resolve such effectiveness bottlenecks, several research and proof of concept studies have proposed the use of ML algorithms for fake news detection. These include text-relevant features (e.g. TF-IDF and n-grams), while another approach which seems to be prominent is the use of linguistic features (e.g. words belonging in certain categories, Part Of Speech tags and others) in combination with ML algorithms. The use of linguistic features offers the ground for developing novel tools that can improve the accuracy in fake news detection (Horne & Adali, 2017; Rubin, Conroy, Chen, & Cornwell, 2016).

The above challenges pose many research questions and the complexity of the problem demands novel and solid solutions. This work is motivated by the next three crucial research questions:

RQ1: *Which linguistic features lead to high accuracy results in fake news detection?*
RQ2: *Would the combination of word embeddings and linguistic features improve the performance in the fake news detection task?*
RQ3: *Which ML algorithm is the most accurate in fake news detection over several datasets?*

To respond to the above RQs, we propose an effective approach for the evaluation of linguistic feature sets and word embeddings text representation along with ML algorithms over several datasets.

In summary, the main contributions of this work are the following:

- **perform an extensive feature set evaluation study** with the purpose to lead in an effective feature set to detect fake news articles. By advancing state of the art of feature sets proposed to detect deception in written narratives as well as word embeddings, we propose a combination of features based on earlier similar studies which are summarized in Section 2.
- **perform an extensive Machine Learning (ML) classification algorithm benchmarking study**, for introducing a robust model that detects fake news articles using the best performing feature set. This study contains multiple ML algorithms well known for their high performance in text classification tasks such as Support Vector Machines (SVM) (Cortes & Vapnik, 1995), K-Nearest Neighbors (KNN) (Cover & Hart, 1967), Decision Trees (DT) (Breiman, Friedman, Stone, & Olshen, 1984), as well as ensemble algorithms such as AdaBoost (Freund & Schapire, 1997) and Bagging (Breiman, 1996).
- **set certain rules and a solid methodology for creating an unbiased dataset for Fake News detection** and create the UNBiased dataset containing a balanced number of false and real news articles. We shall call a dataset unbiased if it contains a balanced distribution of articles from different categories and from different news sources. Fake news articles are filtered using top Fact Checking organizations as listed by [7,8]. An unbiased real news dataset has also been generated by obtaining articles from different topics and various credible journalism organizations. It must be emphasized that all those articles fall into several subject categories such as politics, life & style, sports and more. That is because we aim to produce a generic dataset for fake news detection avoiding bias for a specific topic or editor. This collection was motivated by the lack of publicly available datasets that contain human annotated fake news and plurality in real news content and sources.
- **quality results in fake news detection.** Our experimentation has shown that our approach can achieve accuracy up to 95% over five fake news detection datasets.

The rest of this paper is organized as follows: Section 2 has a review of related work about linguistic features and deception detection. Then, Section 3 presents the feature set and the model selection pipelines along with basic tools used for the evaluation process of each step. Next, in Section 4 we present available datasets for fake news detection and we describe the proposed approach for creating an UNBiased dataset while Section 5 outlines the experimentation results. Finally, in Section 6, we summarize conclusions and we propose future work plan.

2. Related work

As explained above, among the "fakesters" many faces, deception is the most intense and prominent in the state of the art. Next, a summary of related work is categorized with respect to the deception's linguistic profiling, and the ML efforts dealing with the problem. In paragraph 2.1 we present studies that introduce linguistic features for detecting deception in written narratives. In paragraph 2.2 we present studies that address the need of an automated tool for fake news detection and try to exploit linguistic features for creating one.

2.1. Deception linguistic features and profiling

A popular approach which gained prominence in the mid-2000s was the use of linguistic cues for detecting deception in written narratives. Experiments performed by psychologists in cooperation with linguistics experts and computer scientists, revealed that the potential deceivers use certain language patterns, such as small sentences, lots of phrasal verbs, certain tenses etc. (Burgoon, Blair, Qin, & Nunamaker, 2003; Hancock, Curry, Goorha, & Woodworth, 2007; Newman, Pennebaker, Berry, & Richards, 2003; Tausczik & Pennebaker, 2010; Zhou, Burgoon, Nunamaker, & Twitchell, 2004). More specifically, Burgoon et al. (2003) have tested 16 linguistic features that could help discriminate deceptive communications from truthful ones. In order to create a database, they ran two experiments in which either Face to Face or Computer Based discussions were set up with the one participant playing the role of the deceiver while the other one was sincere. Then, the discussions were transcripted for further analysis and the authors con-

[3] https://apnews.com/2da20dba67b14a22b016ed5b94636bc0.
[4] https://www.snopes.com/.
[5] https://www.politifact.com/.
[6] https://www.truthorfiction.com/.
[7] http://www.poynter.org/fact-checkers-code-of-principles/.
[8] https://newsroom.fb.com/news/2016/12/news-feed-fyi-addressing-hoaxes-and-fake-news/.

Table 1
Linguistic features as proposed by Burgoon et al. (2003).

Category	Description
Quantity	# syllables # words # sentences
Vocabulary Complexity	# big words # syllables per word
Grammatical Complexity	# short sentences # long sentences Flesh Kincaid grade level avg # of words per sentence sentence complexity number of conjunctions
Specificity and Expressiveness	emotiveness index rate of adjectives and adverbs # affective terms

cluded to specific linguistic cues classes that could reveal the deceiver. In order to cluster and obtain a hierarchical tree structure of the features proposed, they used the C4.5 Decision Tree algorithm with 15-fold cross-validation. The overall accuracy of their method reached 60.72% in a small dataset of 72 instances. The features proposed is separated into four categories, namely Grammatical Complexity, Vocabulary Complexity, Quantity and Specificity/Expressiveness as presented in Table 1.

Similarly, Newman et al. (2003) have also aimed at a multivariate linguistic profile of deception. More specifically, they proposed a set of five out of twenty-nine linguistic cues which is an intersection of the most significant predictors of deception. This intersection is a result of a systematic analysis of five different experimental case studies. Each case study had different context, a different number of participants and different distribution between sexes (male, female) i.e. "Typed abortion attitudes" with 44 participants (18 male, 26 female), "Mock crime" with 60 participants (23 male, 27 female) etc. In each case study participants were asked to be either deceptive or sincere. The authors used Logistic Regression (LR) for feature evaluation with the algorithm to obtain better results than human judges with 67% vs 52% of accuracy, respectively. Table 2 presents the total features studied which fall into three main categories, namely Psychological Processes, Standard Linguistic Dimensions and Relativity.

Zhou et al. (2004) have also build a linguistic profile of deception. They proposed a set of twenty-seven linguistic features that are clustered into nine categories. To obtain a dataset for evaluating their features, they performed an experiment based on the Desert Survival Problem (Lafferty, Pond, & Synergistics, 1974). In that scenario, the participants used a web-based messaging system for information exchange. As in (Burgoon et al., 2003), participants were separated in pairs where one had the role of deceiver while the other was acting sincerely. Then the authors performed statistical analysis for the evaluation of the features with the results proving the viability of using linguistic-based cues to distinguish truthful from deceptive messages. In this study, the authors separated the features in more categories as presented in Table 3.

Table 2
Linguistic features as proposed by Newman et al. (2003).

Category	Description
Standard Linguistic Dimension	Word Count % Words captures, dictionary words % Words longer than six letters % Total Pronouns % First Person Singular % Total First Person % Total Third Person % Negations % Articles % Prepositions
Psychological Processes	Affective or emotional processes Positive emotions Negative emotions Cognitive Processes Causation Insight Discrepancy Tentative Certainty Sensory and Perceptual Processes Social Processes
Relativity	Space Inclusive Exclusive Motion Verbs Time Past tense verb Present tense verb Future tense verb

Table 3
Linguistic features as proposed by Zhou et al. (2004).

Category	Description
Quantity	# Words # Verbs # Noun Phrases # Sentences
Complexity	avg # clauses avg sentence length avg word length avg noun phrase length Pausality
Uncertainty	Modifiers # Modal Verbs # Uncertainty # Other reference
Non Immediacy	Passive Voice Objectification Generalizing Terms Self Reference Group Reference
Expresivity	Emotiveness Lexical Diversity Content Word Diversity Redundancy Typographical error ratio
Specificity	Spatio-temporal information Perceptual Information
Affect	Positive Affect Negative Affect

2.2. Detecting fake news with linguistic features and machine learning

The use of linguistic cues and ML approaches for fake news detection has gained attention in earlier work such as in (Horne & Adali, 2017) who have used linguistic cues and proposed an SVM classifier with a linear kernel. The authors faced fake news detection as a multi-class problem and they tried to identify whether an article falls into Real, Fake or Satire category. They achieved a 78% of accuracy after a 5-fold cross validation in Fake vs Real news classification while they concluded that satire pieces have very similar characteristics with fake news. They performed their experiments in a dataset provided by Buzzfeed's author Craig Silverman[9] that was enriched with satire articles. The final form

[9] https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook.

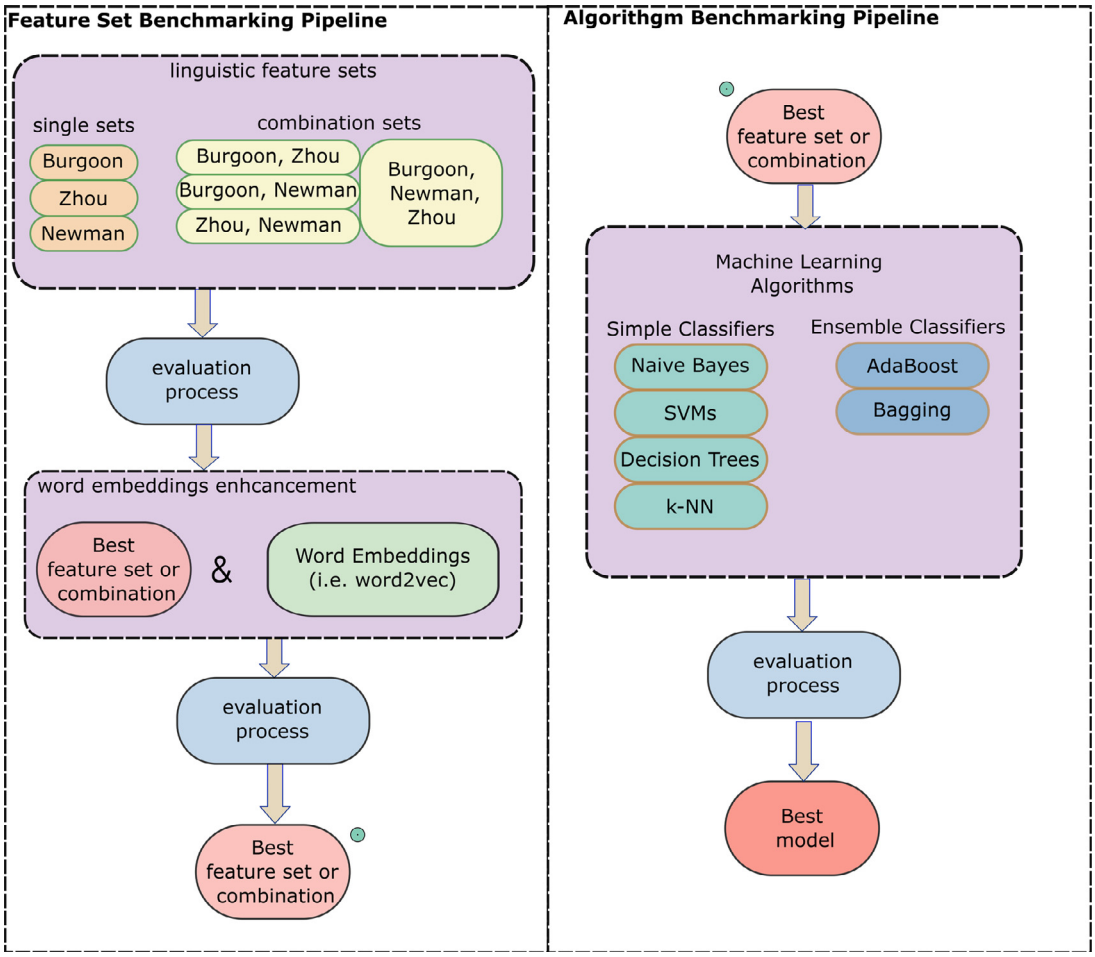

Fig. 1. Benchmarking pipelines followed in this study.

of their dataset consisted of 225 articles equally distributed in the three classes (75 articles in each), while the feature set they used contained mainly POS tags and some Linguistic Inquiry Word Count (LIWC) (Tausczik & Pennebaker, 2010) word categories. LIWC is a dictionary developed to capture basic emotional and cognitive dimensions originally studied in social, health, and personality psychology while lately more word categories have been added by the authors.

Also, several survey articles (Chen, Conroy, & Rubin, 2015; Conroy, Rubin, & Chen, 2015; Rubin, Chen, & Conroy, 2015), have underlined the need for an automated fake news detection tool, which has been demonstrated at (Rubin et al., 2016). The results of this last study are promising since they achieved 90% precision 84% recall and 87% F-score with a Linear SVM and 10-fold cross-validation. Moreover, the authors introduce five satirical cues for detecting potential misleading news and they evaluate their proposed features in an equally distributed dataset of 360 satire and real news articles.

Moreover, Ahmed, Traore, and Saad (2017) used n-grams and TF-IDF for feature extraction in combination with ML techniques to detect fake news. They compared k-NN, SVM, LR, DT and Stochastic Gradient Descent, with the Linear SVM achieving best results with 92% accuracy after a 5 fold cross-validation. In this work, the authors used the Kaggle's fake news dataset[10] which consists of approximately 12600 fake news articles. The authors enriched it with 12600 real news articles collected from Reuters, so a single source of data was used.

From the previous work discussion, it is important to note that there is a lack of benchmarking between ML methods. Both Horne and Adali (2017) and Rubin et al. (2016) propose a Linear SVM, while Ahmed et al. (2017) try more algorithms but not ensemble ones in a single dataset. Moreover, we identify the inconsistency between the datasets proposed and used until now for fake news detection. Horne used a dataset containing 225 articles equally distributed in three classes namely real, fake and satire, while Rubin proposes a 360 articles dataset balanced between real and satire news. Ahmed uses an enhanced Kaggle's dataset with real news, thus he obtained real news only from one source (Reuters) and the fake news were not manually annotated. We believe that this approach may lead in a biased dataset because each journalism organization employs news editors who follow specific style guides.

3. Feature set and model selection

Taking under consideration the related work presented in Section 2 we further exploited the use of linguistic-based features in combination with ML methods to detect news with deceptive content. In this section, we present the *benchmarking pipeline* followed for feature set and ML algorithm selection. In the first part (left in Fig. 1) we present the process we follow to evaluate linguistic feature sets and the word embedding enhancement. Moreover, we present our *evaluation process*, we define *feature sets* and we investigate how to employ *word embeddings* to en-

[10] https://www.kaggle.com/mrisdal/fake-news.

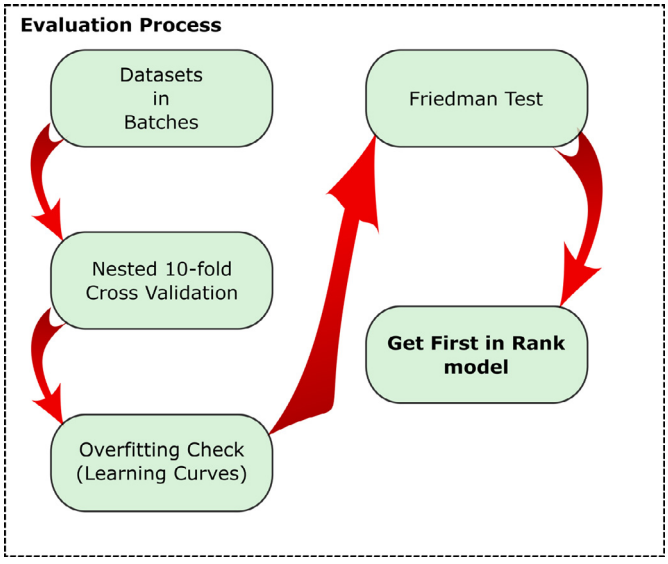

Fig. 2. The evaluation process we used for this study.

hance results. We select our best-performing feature set based on classification accuracy of a baseline SVM linear model. Then we briefly describe a rich set of *classifiers* used for the algorithm benchmarking pipeline (right on Fig. 1). The result of the feature set and algorithm benchmarking pipelines is an accurate model for fake news detection task.

3.1. Evaluation process

In order to ensure the equivalent evaluation of all feature sets and algorithms, we followed a certain Evaluation Process (EP) as depicted in Fig. 2. This process consists of several steps which are ensuring the reliability of results for the algorithms and feature sets benchmarking. To improve the validity of our tests, we used a variety of Fake News datasets available online, as well as our own unbiased (UNBiased) dataset we created from a number of news sources as described in Section 4. The statistical significance of ranking results is further exploited with the Iman and Davenport (Iman & Davenport, 1980) version of Friedman test as explained in detail in paragraph 3.1.2.

3.1.1. Datasets in batches and cross-validation

Since in this study we intend to conclude in a high-performance model capable to detect fake news in real life tasks, we try to keep our pipeline as objective as possible. To achieve such a goal, we use several datasets for the evaluation process, which are of different sizes. To ensure the equivalent evaluation of all models, we split all datasets into random batches of up to 1000 instances with the two classes (i.e. fake and real news) to be equally distributed. The splitting of the datasets into batches was done in order to increase the number of experiments to efficiently support the Friedman Statistical test. We then implement a nested 10 fold cross-validation in each dataset batch. More specific we first split each batch in a 70% train and 30% test set. Then we perform a 10 fold cross-validation in the train set and we test the best parameters in the rest 30% of each batch. Since the two classes are equally distributed in each batch, we use accuracy as a performance indicator. Moreover, we refer to learning curves plots to avoid over-fitting effect and get the optimum parameter values.

3.1.2. Friedman statistical test

According to Demšar (2006), Friedman Statistical Test (FST) with its improvement by Iman and Davenport (1980) is a state-of-the-art statistical test for comparing ranking significance of multiple methods over several datasets. In our study, we leverage the aforementioned test to evaluate the following:

- The **linguistic feature sets** and the **combinations** of them as described in 2.1.
- The **best** performing feature set or combination of the first evaluation enhanced with **word embeddings** and the **combination** of them.
- Several **classification algorithms**.

with all of them to be cases that can be handled with FST.

For the implementation of FST, we have to calculate x_F^2 as described by the following equation:

$$x_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{1}$$

Let r_i^j be the rank of the $j-th$ of k methods on the $i-th$ of N dataset batches. The FST compares the average ranks of algorithms as calculated by Eq. (2).

$$R_j = \frac{1}{N}\sum_i r_i^j \tag{2}$$

The null-hypothesis here states that all methods are equivalent so their ranks Rj should be equal. When the number N of dataset batches and the number of methods k to be compared are big enough the Friedman statistic is distributed according to x_F^2 with $k-1$ degrees of freedom. R_j^2

As explained in (Demšar, 2006), Iman and Davenport (1980) introduced a better statistic, not too conservative as Friedman's test, which is distributed according to the $F-distribution$ with $k-1$ and $(k-1)(N-1)$ degrees of freedom and described by the following equation:

$$F_F = \frac{(N-1)x_F^2}{N(k-1)-x_F^2} \tag{3}$$

After calculating Eqs. (1) and (3) we compare results with the critical values that can be found in statistical tables. If the null-hypothesis is rejected, we proceed with Nemenyi post-hoc test to compare all methods to each other. This post-hoc test reveals if the performance difference of two methods is statistically significant (i.e. if the average rank distance between two methods is at least equal with the critical difference calculated with Eq. (4) the difference between them is not statistically significant).

$$CD = q_a\sqrt{\frac{k(k+1)}{6N}} \tag{4}$$

where q_a is based on the Studentized range statistic divided by $\sqrt{2}$. For this study, we have set $p \leq 0.05$. All the variables of this paragraph are further described in Table 4.

Table 4
Variables used in statistical test.

No	Variable	Description	Equation
1	x_F^2	Friedman statistic	1
2	N	# of datasets	1
3	R_j	average ranks of methods	1
4	k	# of models to be compared	1
5	F_F	Iman & Davenport statistic	3
6	CD	Critical Difference	4
7	q_a	critical value	4

3.2. Feature set selection

As described in 2.1 several studies propose the use of partially different linguistic features for detecting deception in written narratives. For this study, we define a *feature set* to be a list of features

Table 5
Feature set representation for this study.

Feature set proposed by	Representation
Burgoon et al. (2003)	**[a]**
Newman et al. (2003)	**[b]**
Zhou et al. (2004)	**[c]**

as proposed by a relevant published study for deception detection. Moreover, we performed individual feature selection using well established techniques such as *Mutual information* (Kraskov, Stögbauer, & Grassberger, 2004) which is further described in 3.2.1 and mRMR (Peng, Long, & Ding, 2005). We found that feature selection using Mutual Information gives better performance than mRMR and therefore we present only Mutual Information in this Section.

The feature sets we use in this study are the features proposed by Burgoon et al. (2003); Newman et al. (2003); Zhou et al. (2004) and are further described in Tables 1, 2 & 3 (paragraph 2.1 while for the individual feature selection task we used the union of the above mentioned feature sets.

3.2.1. Mutual information

Mutual information is the measurement between two random variables X and Y, that shows the dependency between the two variables. Essentially what is measured is how much of the information of a random variable is obtained by observing the other random variable. The value is always non-negative and the higher the value the bigger the dependency between the values. In case the value is 0 then these 2 variables are completely independent.

In this study we calculated mutual information for all features per dataset and we trained models with the top x with $step = 5$ features in range [5,58] as well as for all of them (58 features). In each case, we performed a nested 7 fold cross-validation to obtain best algorithm parameters in 70% of the dataset and we tested each model in the rest 30%.

3.2.2. Feature selection as sets

Since in all of the previously mentioned studies there are signs that different combinations of features are effective for describing texts of various topics for being deceptive, we performed an extensive study of all possible unions of feature sets in pairs to achieve best fake news detection accuracy. In order to evaluate the results and rank the feature sets based on their ability to describe fake news, we used an SVM with a linear kernel as a baseline classifier known for its good results in text classification tasks and accuracy as a performance metric. From this point and for the rest of this study we follow the representation of feature sets as described in Table 5.

3.3. Word embeddings and enhancement

Word embeddings are dense word representations in a low - dimensional vector space. They are used for representing words and/or sentences in many Natural Language Processing tasks and are particularly useful because they can be used directly as input to neural network language models. Traditionally, when processing natural language, the words are represented as distinct numbers, for example, the word *apple* can be stored as *123*. However, when representing a word like this we get no syntactic or semantic information. This leads to models not being able to use information relating words to each other leading to suboptimal word representations and usually means that we may need more data to successfully train statistical models. Word embeddings solve this problem by representing words of similar meaning in close proximity in the representation space.

Word2vec, as proposed by Mikolov, Sutskever, Chen, Corrado, and Dean (2013), is a group of highly efficient computational models for learning word embeddings from raw text. These models are shallow neural networks of two layers that are trained with supervision using a large volume of text and produce vector representations usually in a vector space of several hundred dimensions. Every single word in the text is mapped to a vector in this space, words that are of similar meaning are in close proximity in this space.

For this study we used word2vec since it has been noted in (Mikolov, Chen, Corrado, & Dean, 2013) that such word representation can be used for verification of correctness of existing facts. Apart from word2vec, other approaches for word representation in vector space has been proposed recently with GloVe (Pennington, Socher, & Manning, 2014) being one of the most prominent cases. In this study, we have experimented with GloVe as well but without noticing important differences in performance.

As a next step, we enhanced the best linguistic feature set that came up after the post hoc test in Friedman results with word embeddings. For treating equally the evaluation process we followed the exact same procedure with the feature set benchmarking process as described in detail in 3.2, with the feature sets compared after enhancement to be *word embeddings, best performing linguistic features combination* and, *word embeddings & best performing features combination*.

3.4. Classifier selection

The selection of best performing classifier is very important for creating an accurate fake news detection model. In this study, we compare four classifiers, namely Naive Bayes, SVMs, Decision Trees, k-NNs together with two ensemble methods (AdaBoost and Bagging). The first four classifiers are very popular in text mining problems with Naive Bayes usually to provide an accuracy baseline while SVMs outperforming in such tasks. The ensemble methods are in general the combination of many “weak” classifiers with a purpose to generate a robust one with improved classification performance (Rokach, 2010). The main idea behind AdaBoost is to focus on patterns that are harder to classify while Bagging is one of the most used and high performing ensemble algorithms.

The final step of our approach is the selection of best performing classification algorithm. In order to conclude with the best models for fake news detection, we performed an extensive algorithm benchmarking study following the EP described previously in paragraph 3.1.

4. News data sources for fake news detection

Since “fake news” has become a topic of great interest, several datasets have been published lately. Most of them contain news articles but different standards have been followed concerning the news annotation process. For example, Kaggle’s fake news dataset is based on BSDetector[11] tool which uses a list of “fake news” sources. Subsequently, if one source which was marked as a fake news publisher publishes an article with legitimate content this would be classified as fake.

McIntire has created a relatively large dataset which contains both fake and real news articles. For the Fake news class, McIntire used a part from the Kaggles fake news collection, while for the real news class he obtained articles from credible journalism organizations such as the New York Times, Wall Street Journal, Bloomberg, National Public Radio, and the Guardian that were published between 2015 and 2016.

[11] http://bsdetector.tech/.

Table 6
The main characteristics of the datasets available about fake news.

Descr.	Kaggle-EXT	McIntire	BuzzFeed	Politifact	UNB
annotation by piece	-	-	✓	✓	✓
several real news sources	-	✓	✓	✓	✓
real news topic diversity	✓	✓	-	-	✓
balanced	✓	✓	✓	✓	✓
instances	23340	6310	240	182	3004

Shu, Sliva, Wang, Tang, and Liu (2017a) created two datasets namely the BuzzFeed and the Politifact with the annotation process of news to have been done by journalists. Both datasets are of similar size and context. Apart from headline and body text, news meta-data (e.g. publish data, images, user relationships) are included as well.

For this study, we used all available datasets for fake news detection in order to include all approaches concerning the data collection process. Moreover, we will try to highlight the necessity of following certain rules in order to build an unbiased dataset for training an algorithm that can generalize well in fake news detection task. Thus, we created the UNBiased dataset with respect to the rules we have posed.

Table 6 describes the main characteristics and the size of each dataset used in this study. The language used in all articles of all datasets is English.

4.1. UNBiased Fake news dataset

Since in the proposed approach we use linguistic features for fake news detection, we believe that using a small number of sources and topics either for fake or real news collection may lead to a dataset which is *biased*. Our experiments showed that by training an algorithm on such a dataset may lead in a model which is biased towards that particular source. Moreover, the model may not generalize well because the same source tends to use the same linguistic features other sources may not.

In order to overcome such drawbacks, we created an *unbiased* dataset of satisfactory size in order to evaluate our work. For this process, we have set certain standards and rules for the creation of such dataset which in summary are:

- Each fake news article should be annotated by experts
- Fake news should originate from several sources
- Real news must be published by credible journalism organizations
- Obtain articles of several news categories in order to create a pluralistic collection of real news.

Following the above, we chose not to treat a news source that once has published a piece of news annotated as fake as a “fake news source”. Instead, we have trusted credible fact-checking organizations such as Snopes.com, TruthOrFiction.com and Politifact.org that are treating each piece of news individually. Next, we retrieved the content of the news as they were originally published. Afterwards, we removed duplicates and we cleansed the dataset from articles containing only images. In order to enrich our dataset with Real news, we retrieved stories from several accurate and credible Journalism organizations. For ensuring the elimination of bias factor we collected Real news from several categories such as travel, technology, sport, life & style etc. Moreover, we included opinion based articles from several journalism blogs. We believe that opinion based journalism is trending nowadays, so studying articles written in that motive will become the keystone in fake news detection.

Following the above rules, we created a novel dataset that contains:

- 1400 articles, each one of them annotated by experts as fake news
- 2004 articles from various credible sources with Real news from different categories.

The process of UNBiased creation is depicted in Fig. 3.

4.2. Kaggle-EXT (Kaggle with Reuters)

As described in the begging of this section, Kaggle has published a dataset of approximately 12600 news articles annotated as fake by the BSDetector tool which takes into account the source of news.

To get closer to (Ahmed et al., 2017), we enhanced this dataset with the Reuters published news corpus as is in NLTK data collection (Loper & Bird, 2002). The result was a large and balanced dataset concerning the two classes i.e. “Kaggle-EXT”.

However, as we have already noticed, we have certain doubts about the annotation process of fake news and the bias factor for real news, since they originate only from one source (i.e. Reuters).

4.3. McIntire

As denoted in the introduction of this section, McIntire used a part of Kaggle’s fake news dataset and enhanced it with news from several credible journalistic organizations. We believe that this approach is leaning towards the creation of an unbiased dataset, but yet, it does not conform to the standards we have posed about the fake news annotation process since fake news articles are not annotated individually and not by humans. However, the size of this dataset satisfies the needs for an extensive text classification problem as this study is. The dataset is available online[12].

4.4. Kaidmml

The authors of (Shu et al., 2017a) and (Shu, Wang, & Liu, 2017b) created two datasets in order to perform their studies about fake news detection. They collected articles annotated by two credible news organizations namely Politifact and Buzzfeed. The annotation process conforms the standards we have posed above, with the only disadvantage being their size, which is relatively small for training a machine learning algorithm.

5. Experimentation - Results

In this section, we present the results of the proposed evaluation process. For each step, suitable diagrams are presented in order to easily select the optimum parameters and move to the next step of the benchmarking pipeline.

5.1. Linguistic feature sets evaluation results

As described in paragraph 3.2, for concluding in the best linguistic feature set for fake news detection, we followed certain

[12] https://github.com/GeorgeMcIntire/fake_real_news_dataset.

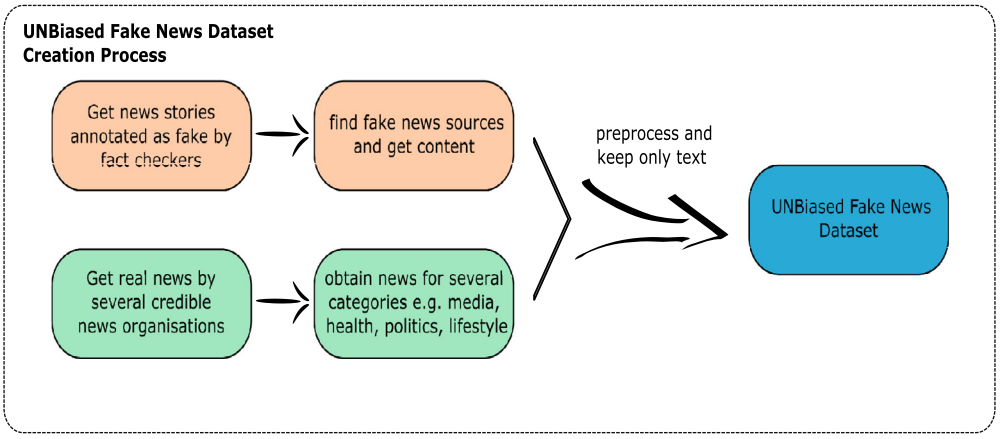

Fig. 3. UNB Dataset creation process. From the news articles obtained we keep only the text.

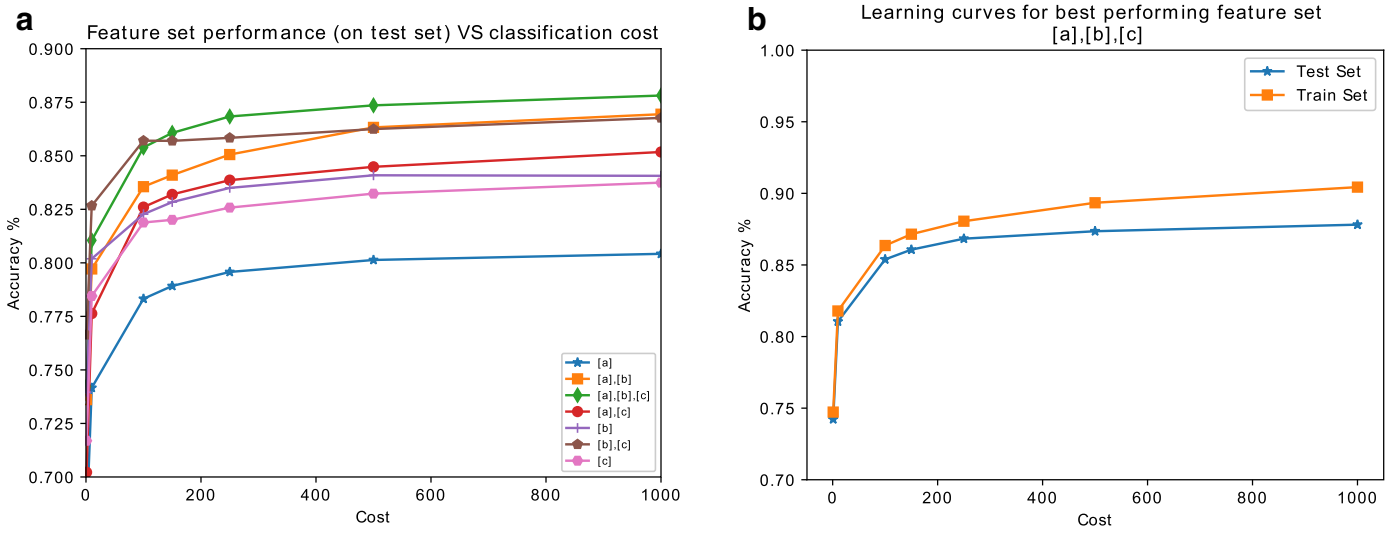

Fig. 4. (a) Performance of each feature set or combination vs C parameter. (b) Learning curves for best performing feature set.

steps. First, we tried to find which feature set performs best in general. For each dataset batch, we trained an SVM with a linear kernel using the 70% of the data and tested in the rest 30%. We repeated the above procedure for each feature set combination and for several values of *Cost* parameter namely [1, 10, 100, 150, 250, 500, 1000]. The *Cost* parameter indicates the cost of misclassification in SVM algorithm. The results of this first step are depicted in Fig. 4a where it is clear that the combination of *a, b & c* feature sets achieves the best accuracy over all other possible combinations. Then, in order to select the optimum cost value and avoid over-fitting, we plotted the learning curves for the best feature set combination as shown in Fig. 4b. With respect to a balanced decision in between bias and variance factors, we choose to continue with the Friedman test for a value of $Cost = 250$. The test results showed that we had to reject the null-hypothesis so we performed the Nemenyi post hoc test in order to evaluate the performance of each feature set and combination. The accuracy and ranking results are presented in Table 8.

For illustrating the results of Nemenyi post hoc test, we adopted the approach proposed by (Demšar, 2006). In this illustration when the vertical lines are cut by a horizontal one there is no statistically significant difference between them. As depicted in Fig. 5, the combination which performs best is composed by the union of feature sets proposed by Burgoon et al. (2003); Newman et al. (2003); Zhou et al. (2004) depicted as **[a], [b], [c]** respectively while the second with no statistically significant difference is the combination of **[b], [c]**. Another observation is that when feature sets are used individually perform worse than when used in combinations. For the rest of our study, we accept the union of **[a], [b], [c]** as the best performing linguistic feature set for fake news detection. The above set is further described in Table 7.

5.2. Word embeddings enhancement evaluation results

As a next step, we enhanced the best linguistic feature set with features calculated with the state-of-the-art word2vec method. We used a pre-trained (with the Google News corpus) model in order to extract a 300-word embeddings vector for each article. Then, we proceeded with the evaluation process of the proposed approach as described in 3.3. Fig. 6a depicts the *performance curves* of the feature sets while Fig. 6b illustrates the *learning curves* for the best combination which is the enhanced with word embeddings, [a], [b], [c] linguistic feature set (i.e. **[a], [b], [c], [embeddings]**). The results of this step are quite interesting since it is clear that word embeddings boost the performance of the trained model. Yet, according to the learning curves plot (Fig. 6b), we have to set $cost = 150$ in order to avoid overfitting. As a final step of the process, we performed the FST to find if there is statistically signif-

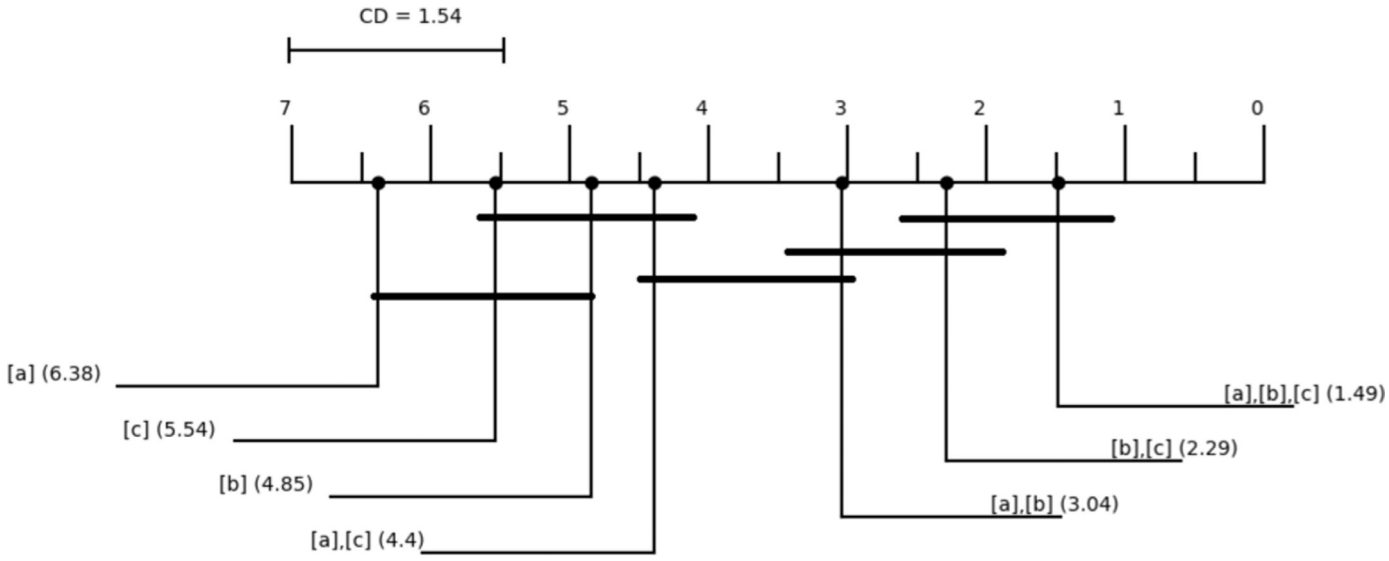

Fig. 5. Nemenyi post hoc test for feature sets benchmarking.

Table 7
Best performing linguist feature set combination.

A/A	Description	A/A	Description
1	# syllables	30	Tentative
2	# words	31	Certainty
3	# sentences	32	Sensory and Perceptual Processes
4	# big words	33	Social Processes
5	# syllables per word	34	Space
6	# short sentences	35	Inclusive
7	# long sentences	36	Exclusive
8	Flesh Kincaid grade level	37	Motion Verbs
9	avg # of words per sentence	38	Time
10	sentence complexity	39	Past tense Verb
11	number of conjunctions	40	Present tense Verb
12	emotiveness index	41	Future tense verb
13	rate of adjectives and adverbs	42	# Noun phrases
14	# affective terms	43	avg # clauses
15	% Words captures, dictionary words	44	avg word length
16	% Words longer than six letters	45	avg noun phrase length
17	% Total Pronouns	46	pausality
18	% First Person Singular	47	modifiers
19	% Total First Person	48	# modal verbs
20	% Total Third Person	49	passive voice
21	% Negations	50	Objectification
22	% Articles	51	Generalize Terms
23	% Prepositions	52	Group Reference
24	Positive Emotions	53	Lexical Diversity
25	Negative Emotions	54	Content word diversity
26	Cognitive Processes	55	Redundancy
27	Causation	56	Typographical error ratio
28	Insight	57	Spatio - temporal information
29	Discrepancy		

Table 8
Average accuracy and ranking for each feature set or combination over all batches.

Set	Accuracy	Rank	Set	Accuracy	Rank
[a]	0.796	6.38	[a,c]	0.839	4.4
[b]	0.835	4.85	[b,c]	0.858	2.29
[c]	0.826	5.54	**[a,b,c]**	**0.868**	**1.49**
[a, b]	0.851	3.04	-	-	-

Table 9
Average accuracy and ranking results for best linguistic feature set and word embeddings.

Set	Accuracy	Rank
[a][b][c]	0.861	2.94
[embeddings]	0.937	1.79
[a][b][c][embeddings]	**0.949**	**1.26**

icant difference between the performance of the proposed feature sets. Similar with 5.1, the test showed that we have to reject the null - hypothesis and to proceed with the Nemenyi post hoc test. Fig. 7 depicts that the enhancement of the best linguistic style feature set with word embeddings performs best with the second in ranking feature set to be the word embeddings alone and the critical difference to be in the edge for marking the rank results statistically significant. In Table 9 is presented the classification accuracy obtained with each feature set.

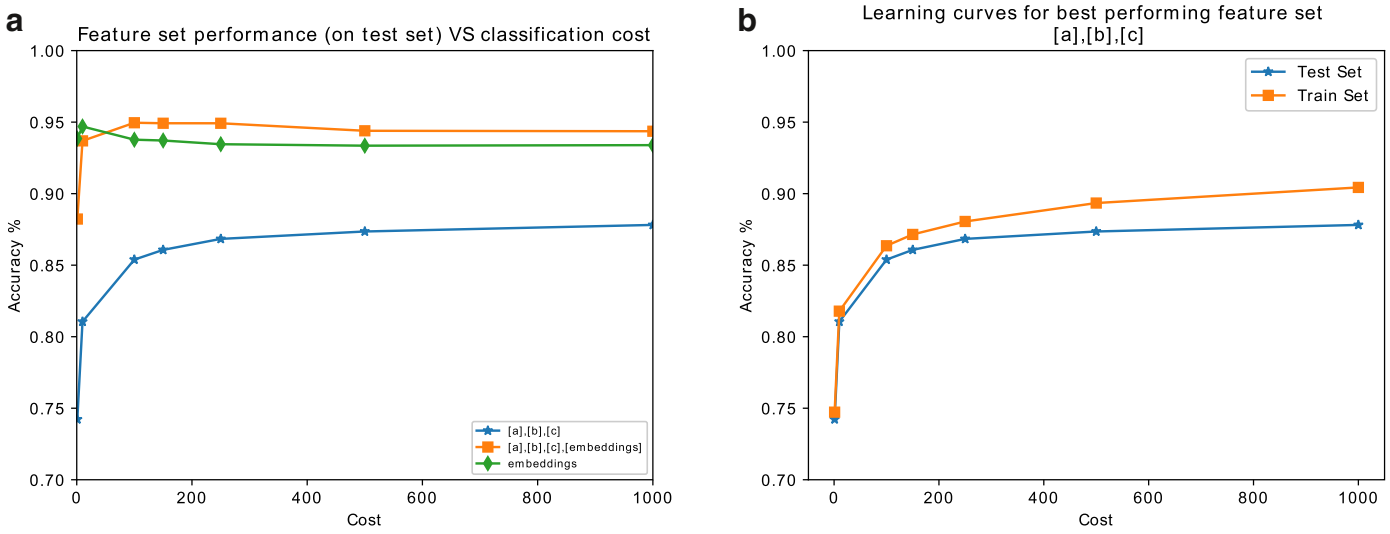

Fig. 6. (a) Performance of each feature set or combination vs C parameter. (b) Learning curves for best performing feature set.

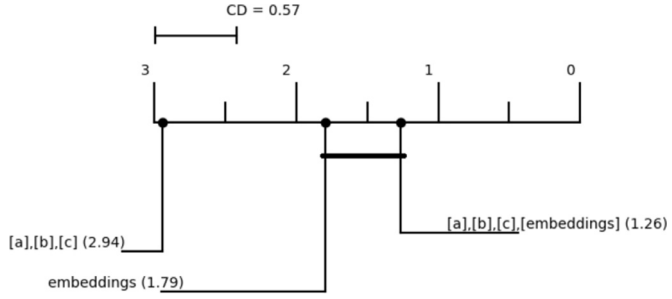

Fig. 7. Nemenyi post-hoc test for linguistic, word embeddings and combination feature sets.

Table 10
Algorithm accuracy results and ranking.

Algorithm	Accuracy	Rank
k-NN	0.921	3.99
Decision Tree	0.858	5.57
Naive Bayes	0.881	5.09
SVM	**0.950**	2.44
AdaBoost	0.949	**1.85**
Bagging	0.944	2.06

5.3. Algorithm evaluation results

After selecting the best feature set combination, we continued our study with an algorithm evaluation. As noted in paragraph 3.4 the final step for deciding on the best model for fake news detection is the performance evaluation of several Machine Learning algorithms, namely SVMs, Naive Bayes, Decision Trees, k-NN and ensemble methods such as AdaBoost and Bagging. In order to keep our study consistent, we followed again the evaluation process described in paragraph 3.1. In Fig. 8 is depicted the ranking of all algorithms used in this study based on their accuracy while in Table 10 is presented the average classification accuracy over all datasets.

5.4. Mutual information with linguistic features

As described in Section 3.2 apart from using feature sets, we calculated Mutual Information of each feature to estimate the discriminatory power they have. It is clear that different features have different discriminatory powers depending on the

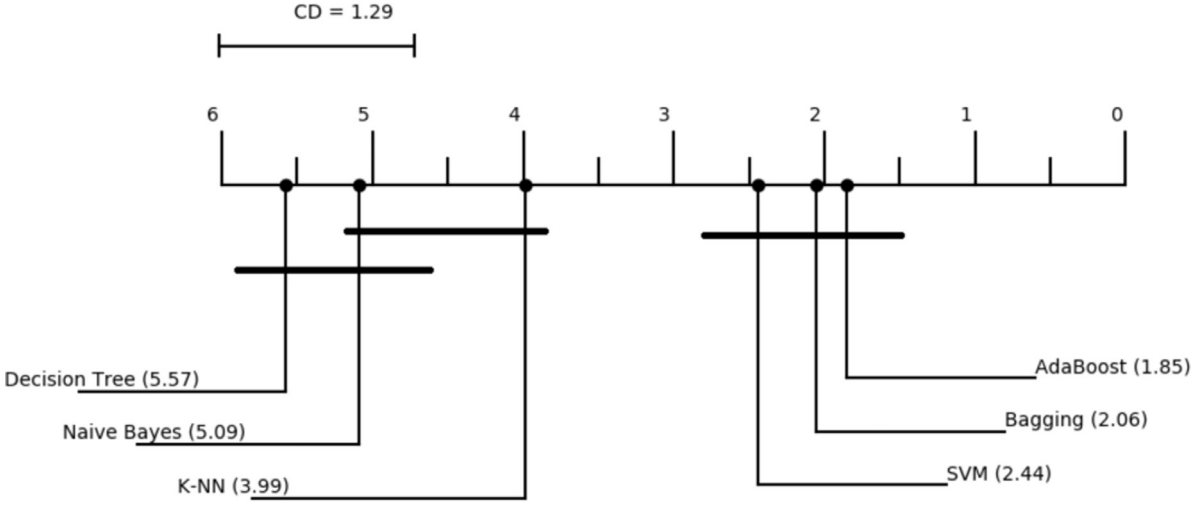

Fig. 8. Nemenyi post-hoc test for all the algorithms tested in this study.

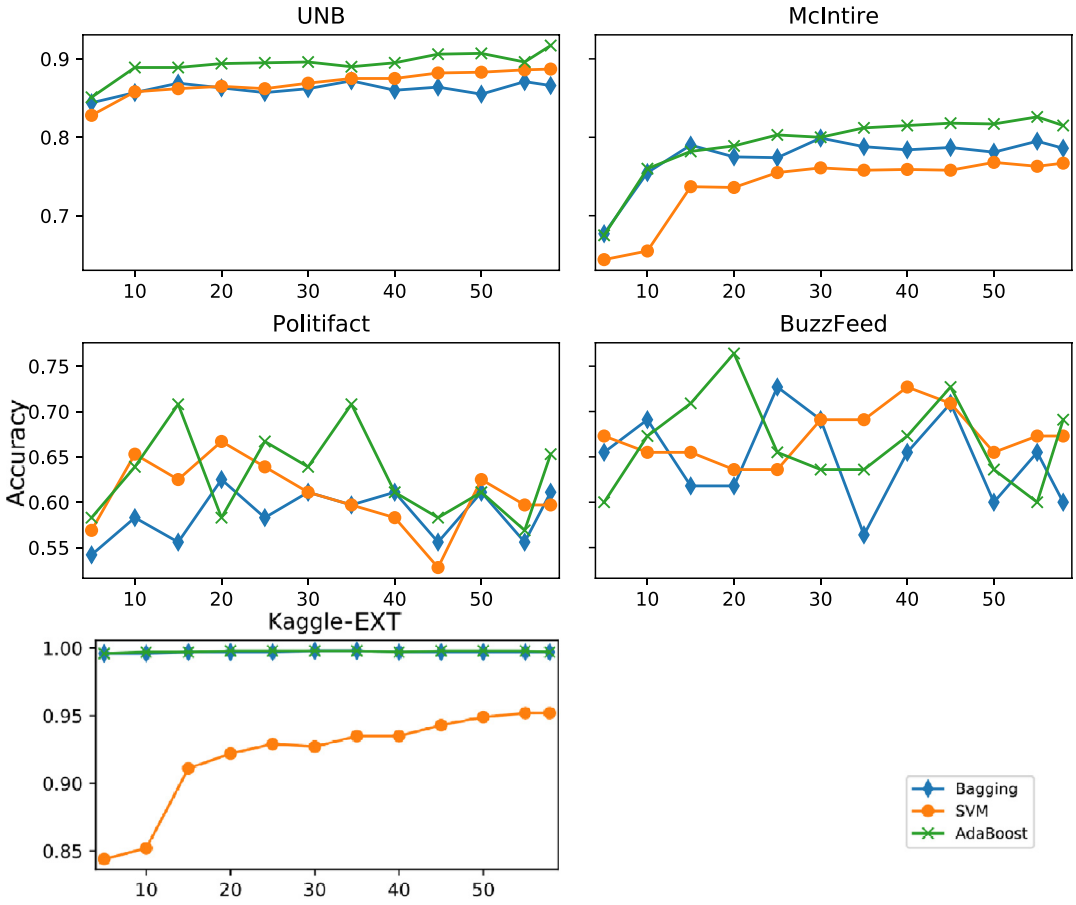

Fig. 9. Accuracy per algorithm and dataset based on feature selection by mutual information. The horizontal axis corresponds to the number selected features.

Table 11
Top ten features based on Mutual Information for each dataset.

ranking	Kaggle	McIntire	BuzzFeed	Politifact	Unbiased
1	Typos	Pausality	Bigwords	Bigwords	Pronoun
2	Redundancy	Longsent	RAA	WC	Wps
3	Shortsent	NP	Focuspast	FK	Social
4	Sentences	WC	Sentencedepth	Space	Redundancy
5	Pronoun	Sentencedepth	Cogproc	Lexicaldiversity	FK
6	WC	Syllables	Pausality	Contentdiversity	Dic
7	Modifiers	Bigwords	Cause	Modifiers	Wordlength
8	Syllables	Contentdiversity	Posemo	NP	Verb
9	NP	Modifiers	Excl	Sentences	RAA
10	Longsent	Otherp	Redundancy	Cause	NP

dataset as shown in Table 11. Yet, we observed that classifiers have similar behavior when tested in datasets that are created with the same principles (e.g. multi-source and multi - domain content).

More specifically, in the “McIntire” and “Unbiased” datasets the classifiers perform best when all features are used. This is because, those datasets are not biased concerning the source or the domain of the news. Subsequently, more features are needed to describe the language, the style and the intention of the “fakester” to deceive the reader. Instead, in “Kaggle-EXT” where the fake news class is complemented with real news from only one source, just one feature (namely *typos*) is enough to get high accuracy results. This proves that “Kaggle-EXT” is not a representative dataset for training and evaluating a Fake News classification model. Moreover, the poor performance in the two small datasets “BuzzFeed” and “Politifact” indicates that dataset size is important for building a robust model.

5.5. Discussion

With a closer look in Figs. 10 & 11 we can conclude to the following: First, in small datasets such as BuzzFeed and Politifact, we observe a large spread of accuracy results for different algorithms. Second, there is a significant difference in the accuracy score between Kaggle-EXT and McIntire dataset results. Since in both datasets the fake news class is the same while what they vary only in the real news sources, we believe that this difference is an indicator of bias risk concerning the use of only one source either for real or fake news articles.

Furthermore, we note that the combination of proposed feature sets for detecting deception in written narratives along with the enhancement with word2vec features does perform best uniformly over five datasets of different characteristics.

Following the algorithm evaluation experiments, we conclude that ensemble algorithms (AdaBoost & Bagging) and SVM perform

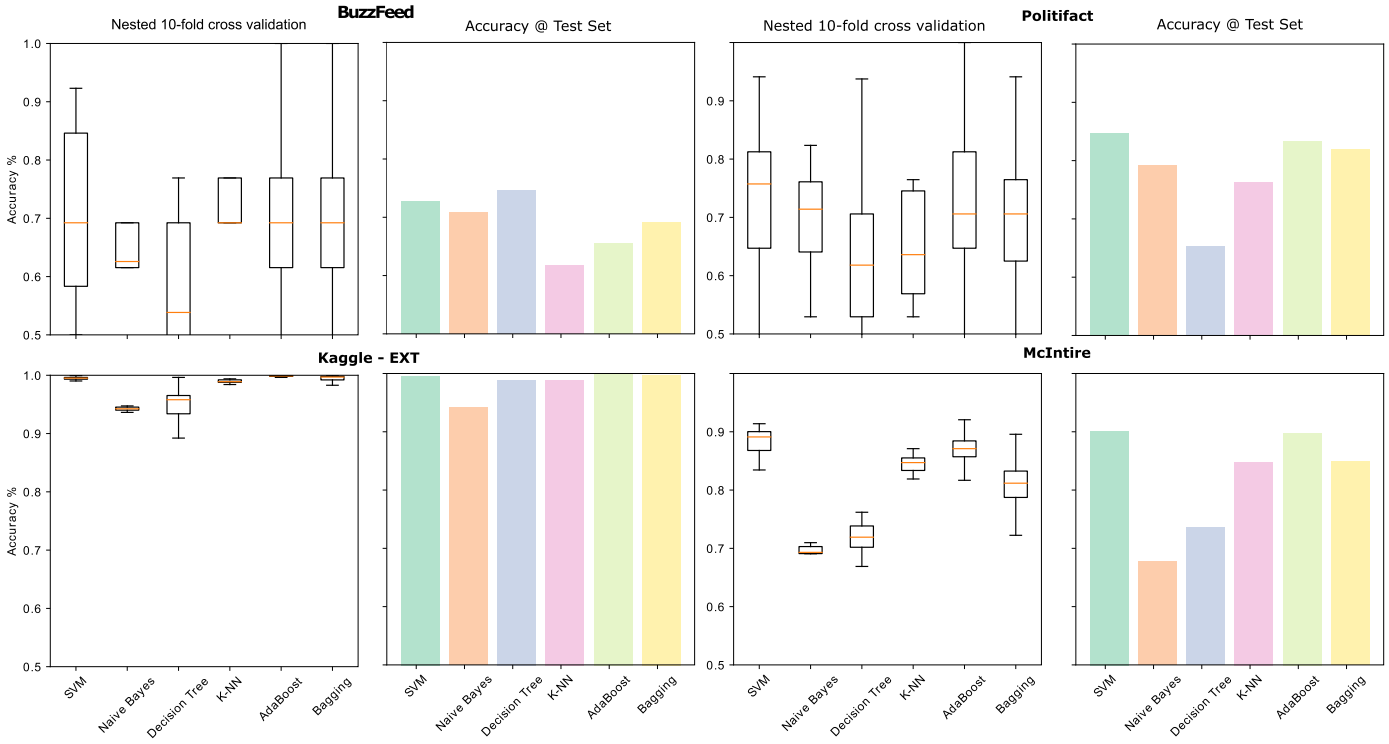

Fig. 10. Algorithm benchmarking over all datasets except for UNBiased Dataset.

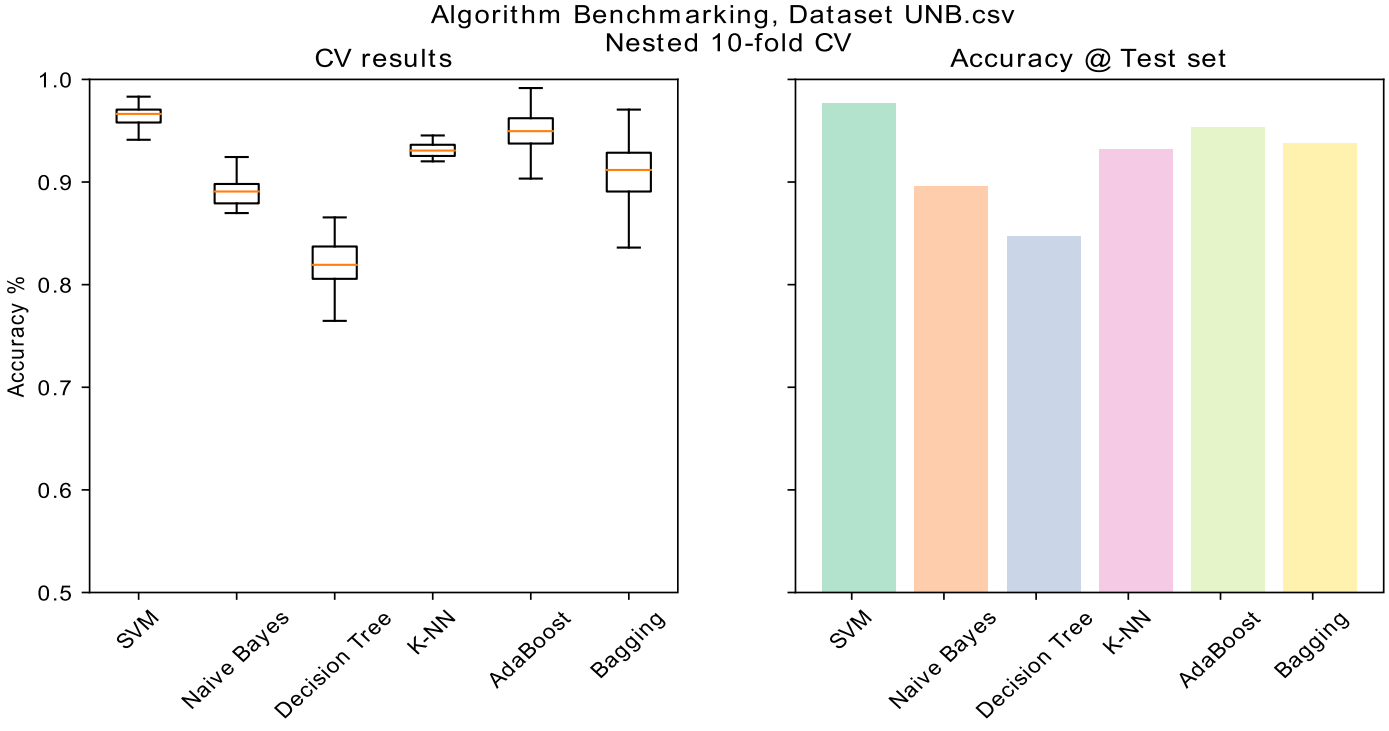

Fig. 11. Algorithm evaluation in UNBiased dataset.

best and are very close in ranking while the other algorithms tested being lower in ranking as depicted in Fig. 8.

It is worth noting that Ahmed et al. (2017) employed TF-IDF features for detecting fake news using the original Kaggle dataset enhanced with 12600 real news from a single source (Reuters). The dataset used by Ahmed is not publicly available, so it was not possible to use it to directly compare our method against their results. Nevertheless, we note that they achieved an accuracy of 92% while our model achieved 99% accuracy in the Kaggle with Reuters dataset. The performance difference may be an indication that the use of linguistic features could be more powerful than term-frequency features.

Shu et al. (2017b) use social engagement features along with partisanship indicators of the publisher and content based features. The classification results of their framework are significantly better in BuzzFeed and Politifact datasets (0.864 & 0.878 accuracies re-

Table 12
Content based studies evaluation.

Dataset	RST	LIWC	ours
BuzzFeed	0.610	0.655	**0.727**
Politifact	0.571	0.637	**0.847**

spectively) compared to our only content-based features approach. Yet, our enhanced linguistic feature set over performs both RST and LIWC content-based approaches as calculated in (Shu et al., 2017b) (Table 12). We have to note that we compare only SVM classification results since the authors used an SVM classifier.

6. Conclusions and future work

In this work, we proposed an enhanced set of linguistic features with powerful capabilities for discriminating fake news from real news articles. We performed an extensive study to conclude in this set while we evaluated several classification algorithms driven by the proposed feature set. We ran experiments on five different corpora and we propose a set of rules and standards for fake news dataset creation.

The proposed features combined with ML algorithms obtained accuracy up to 95% over all datasets used with the AdaBoost to be first in rank and SVM & Bagging algorithms to be next in ranking but without statistically significant difference. Such results prove that the classification of articles according to their truthfulness is possible by selecting proper features and suitable ML algorithms. Moreover, the proposed approach could be the base for a tool helping publishers to quickly decide which article needs further exploitation concerning its veracity.

In the future, a possible improvement would be to employ several meta-data about the source and the author of news, along with social media information diffusion features and use Deep Learning methods with larger datasets. In that way, the fake news detection task would not only be content-based and would improve the prevention of their dissemination in social networks.

Credit authorship contribution statement

Georgios Gravanis: Investigation, Data curation, Writing - original draft, Software, Methodology, Visualization. **Athena Vakali:** Supervision, Writing - review & editing, Methodology, Conceptualization. **Konstantinos Diamantaras:** Supervision, Writing - review & editing, Methodology, Conceptualization. **Panagiotis Karadais:** Software, Writing - original draft.

References

Ahmed, H., Traore, I., & Saad, S. (2017). Detection of online fake news using n-gram analysis and machine learning techniques. In *Intelligent, secure, and dependable systems in distributed and cloud environments* (pp. 127–138). Springer International Publishing.

Allcott, H., & Gentzkow, M. (2017). Social Media and Fake News in the 2016 Election. *Working Paper 23089*. National Bureau of Economic Research. doi:10.3386/w23089.

Breiman, L. (1996). Bagging predictors. *Machine learning, 24*, 123–140.

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. (1984). *Classification and regression trees*. Wadsworth.

Burgoon, J. K., Blair, J. P., Qin, T., & Nunamaker, J. F. (2003). Detecting deception through linguistic analysis. In H. Chen, R. Miranda, D. D. Zeng, C. Demchak, J. Schroeder, & T. Madhusudan (Eds.), *Intelligence and security informatics* (pp. 91–101). Berlin, Heidelberg: Springer Berlin Heidelberg.

Chen, Y., Conroy, N. J., & Rubin, V. L. (2015). News in an online world: The need for an automatic crap detector. *Proceedings of the Association for Information Science and Technology, 52*(1), 1–4. doi:10.1002/pra2.2015.145052010081.

Conroy, N. J., Rubin, V. L., & Chen, Y. (2015). Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology, 52*(1), 1–4. doi:10.1002/pra2.2015.145052010082.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning, 20*, 273–297.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory, 13*, 21–27.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research, 7*, 1–30.

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci., 55*(1), 119–139. doi:10.1006/jcss.1997.1504.

Hancock, J. T., Curry, L. E., Goorha, S., & Woodworth, M. (2007). On lying and being lied to: A linguistic analysis of deception in computer-mediated communication. *Discourse Processes, 45*(1), 1–23. doi:10.1080/01638530701739181.

Horne, B. D., & Adali, S. (2017). *This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news* arXiv:1703.09398.

Iman, R. L., & Davenport, J. M. (1980). Approximations of the critical region of the friedman statistic. *Communications in Statistics - Theory and Methods, 9*(6), 571–595. doi:10.1080/03610928008827904.

Klein, D. O., & Wueller, J. R. (2017). Fake news: A legal perspective. *Journal of Internet Law*.

Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. *Physical review E, 69*(6), 066138.

Lafferty, J. C., Pond, A. W., & Synergistics, H. (1974). *The desert survival situation : A group decision making experience for examining and increasing individual and team effectiveness* (7th). Plymouth, Mich. Human Synergistics. Acompanied by session conductor's Manual (40 p. ill.).

Loper, E., & Bird, S. (2002). Nltk: The natural language toolkit. In *Proceedings of the acl-02 workshop on effective tools and methodologies for teaching natural language processing and computational linguistics - volume 1*. In *ETMTNLP '02* (pp. 63–70). Stroudsburg, PA, USA: Association for Computational Linguistics. doi:10.3115/1118108.1118117.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*. arXiv: 1301.3781.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). *Distributed representations of words and phrases and their compositionality* arXiv:1310.4546.

Newman, M. L., Pennebaker, J. W., Berry, D. S., & Richards, J. M. (2003). Lying words: Predicting deception from linguistic styles. *Personality and Social Psychology Bulletin, 29*(5), 665–675. doi:10.1177/0146167203029005010. PMID: 15272998.

Newman, N. (2017). Journalism, Media, and Technology Trends and Predictions 2017. *Technical Report*. Reuters Institute for the Study of Journalism with the support of Googles Digital News Initiative.

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence, 27*(8), 1226–1238.

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

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review, 33*(1), 1–39. doi:10.1007/s10462-009-9124-7.

Rubin, V., Conroy, N., Chen, Y., & Cornwell, S. (2016). Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the second workshop on computational approaches to deception detection* (pp. 7–17). Association for Computational Linguistics. doi:10.18653/v1/W16-0802.

Rubin, V. L., Chen, Y., & Conroy, N. J. (2015). Deception detection for news: Three types of fakes. *Proceedings of the Association for Information Science and Technology, 52*(1), 1–4. doi:10.1002/pra2.2015.145052010083.

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017a). Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter, 19*(1), 22–36.

Shu, K., Wang, S., & Liu, H. (2017b). *Exploiting tri-relationship for fake news detection* arXiv:1712.07709.

Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology, 29*(1), 24–54. doi:10.1177/0261927X09351676.

Zhou, L., Burgoon, K. J., Nunamaker, F. J., & Twitchell, D. (2004). Automating linguistics-based cues for detecting deception in text-based asynchronous computer-mediated communications. *Group Decision and Negotiation, 13*(1), 81–106. doi:10.1023/B:GRUP.0000011944.62889.6f.