Detecting Fake Reviews: Just a Matter of Data

René Theuerkauf
Martin Luther University Halle-Wittenberg
rene.theuerkauf@wiwi.uni-halle.de

Ralf Peters
Martin Luther University Halle-Wittenberg
ralf.peters@wiwi.uni-halle.de

Abstract

Along with the ever-increasing portfolio of products online, the incentive for market participants to write fake reviews to gain a competitive edge has increased as well. This article demonstrates the effectiveness of using different combinations of spam detection features to detect fake reviews other than the review-based features typically used. Using a spectrum of feature sets offers greater accuracy in identifying fake reviews than using review-based features only, and using a machine learning algorithm for classification and different amounts of feature sets further elucidates the difference in performance. Results compared by benchmarking show that applying a technique prioritizing feature importance benefits from prioritizing features from multiple feature sets and that creating feature sets based on reviews, reviewers and product data can achieve the greatest accuracy.

Keywords: Fake Reviews, Detection, Spam, Benchmarking, Feature Selection

1. Introduction

Customer reviews are part of the purchase decision-making process for products, thus the quality of products is intertwined with the quality of their reviews (Luca & Zervas, 2016). As electronic marketplaces have continued to revolutionize how people gather information before purchasing products, customer reviews have become the most important component of electronic word of mouth (Jiménez & Mendoza, 2013). Such reviews can be easily obtained from digital forums and online shops with rating platforms and thus factor into consumers' assessments of the quality of products (Engström & Forsell, 2018). Simon-Kucher's study involving 6,375 respondents spread across 23 countries revealed that 47% of the surveyed consumers reported regularly consulting product ratings before making purchases, and 71% believed that such ratings are at least somewhat important, if not very important, in making purchase decisions (Simon-Kucher, 2019). Older studies have also shown the relevance of customer reviews in purchase decisions (Gu et al., 2012; X. Li & Hitt, 2010), including that reviews are the most relevant factor after price (Askalidis & Malthouse, 2016). As such, online product reviews affect product reputations (Filieri et al., 2015), sales volumes (He et al., 2020), and merchants' profits (Dellarocas, 2006). In fact, the conversion rate of a product can increase by as much as 270% if it accumulates even a small number of reviews that users can access (Askalidis & Malthouse, 2016).

However, of all customer reviews for given products, the proportion of fake reviews has been estimated to be 16% (Luca & Zervas, 2016), 20% (Schuckert et al., 2016) to 33% (Salehi-Esfahani & Ozturk, 2018). Given the established weight of reviews in purchase decisions and for the success of businesses, fake reviews undermine market efficacy (Hunt, 2015) and, in turn, negatively affect social welfare (Song et al., 2017). A recent study in cooperation with the University of Baltimore puts the cost of online fake reviews in e-commerce at $152 billion in 2020. This is based on an overall global e-commerce web revenue of $4.28 trillion, an assumption that 89% of all global e-commerce web revenue was influenced by reviews and an underlying fake review share of 4% of all online reviews (CHEQ, 2021). To counter that trend, recent research on identifying fake reviews has focused on the use of isolated feature sets (Crawford et al., 2015) but not the simultaneous use of review-based, reviewer-based, and product-based data. To date, only the combination of review-based and/or reviewer-based or product-based feature sets have been discussed for the development of classification algorithm (Asghar et al., 2020) and no comparison of the relevance of using different feature sets has been conducted.

The objective of the paper is to motivate the use of product-based and reviewer-based features, in addition to review-based features, to improve the performance of algorithms designed to detect fake reviews. Furthermore, we investigate which features have the greatest influence on the classification decision. In the process, for the first time it seeks to demonstrate which increases in performance are possible by enriching the inputs of features using different feature sets. Following an adopted process based on CRISP-DM (Chapman et al., 2000) and ASUM-DM (IBM Corporation, 2016), a method was developed to enhance the performance of a

URI: https://hdl.handle.net/10125/103166
978-0-9981331-6-4

system for identifying fake reviews and demonstrated in a software prototype. The results were evaluated by benchmarking and inter-study comparison. A dataset of iOS App Store app reviews marked as fake and non-fake from Martens & Maalej was used to conduct the research. Representatives of the three feature sets of review-based, reviewer-based, and product-based data were derived from the available data, and to illustrate the value of the sequentially additive use of the features, the random forest algorithm was used. Each constellation of features was evaluated individually in consideration of corresponding evaluation criteria and compared with the following constellation. To pinpoint the relevance of the features for classification, the importance of all used features was computed.

As a result, this article contributes evidence showing that simultaneously using the three mentioned feature sets can increase the performance of the classification method, which to the best of the authors knowledge has never been analyzed. The evidence thus implies that analyzing textual data only is no longer sufficient while using classical methods in machine learning. In that context, the influence is quantified with an improvement of the F1 score by 20.89%, in the way that the relevance of the inclusion of further data, is motivated.

Section 2 presents the study's theoretical background and related work on the topic. Next, Section 3 describes the experimental design, including the dataset used and the algorithmic method applied, after which Section 4 discusses the results. In closing, Section 5 summarizes the paper and indicates directions for future research.

2. Theoretical Background and Related Work

2.1. Theoretical Background

This article addresses, opinion spam, as one of four types of spam defined by Jindal & Liu in 2008, which can be further divided into deceptive opinion spam and product-irrelevant spam depending on the damage caused to users (Luyang et al., 2017). On the one hand, deceptive opinion spam represents types of reviews with imaginary opinions written to seem authentic (Ren & Ji, 2017). Spammers using deceptive opinion spam give undeserved positive reviews to promote certain products and/or unjustified negative reviews to damage the reputations of other products (Ren et al., 2014) and, in either case, do not need to have experienced the products (Aslam et al., 2019). On the other hand, product-irrelevant spam refers to non-reviews primarily including irrelevant feedback and advertisements containing no opinion about the targeted products whatsoever (Luyang et al., 2017; Ren et al., 2014). Because people can easily identify and ignore product-irrelevant spam, it poses little threat and is therefore not subject of this paper (Ren et al., 2014).

In the practice of identifying fake reviews, mainly two types of machine learning approaches can be distinguished (Luyang et al., 2017; Wu et al., 2020): supervised approaches, including support vector machines (SVM) (Elmogy et al., 2021), neural networks (Luyang et al., 2017; Sun et al., 2016), and random forests (Lau et al., 2012) as well as unsupervised approaches, including joint probabilistic models (Dong et al., 2018), unsupervised matrix iteration algorithms (Yu et al., 2019), and lexicon-based unsupervised models (Kamalesh & Diwedi, 2015). Using those approaches, research on detecting fake reviews has primarily been performed with three gold-standard datasets (Ren & Ji, 2019): a dataset of 400 reviews of 20 hotels in Chicago from TripAdvisor.com (Ott et al., 2013), a dataset of 5.8 million reviews for products in different product categories on Amazon.com (Jindal & Liu, 2008), and a dataset of reviews of hotels and restaurants from Yelp.com (Fei et al., 2013). Because datasets with real fake reviews are generally unavailable (Naveed et al., 2019), most methods in researches involve examining pseudo-labelled data (Luyang et al., 2017). For instance, the TripAdvisor dataset (Ott et al., 2013) was created by having Amazon Mechanical Turk write deceptive opinion reviews of hotels in Chicago.

In using those algorithms on those datasets, the selection of features to be used as inputs primarily considers three feature sets (Asghar et al., 2020; Crawford et al., 2015). First, review-based features can be bag-of-words, term frequency features, or linguistic inquiries and word count outputs. Second, reviewer-based features are based on the identification of spammers' activity patterns and profile characteristics. Third, product-based features provide information about the popularity of products (e.g., sales rank or average rating). In turn, the combination of different features exerts different effects on the detection of fake reviews (Jindal & Liu, 2008).

2.2 Related Work

The term opinion spam, coined by Jindal & Liu, distinguishes such spam from other traditional types of spam such as email spam and web spam (Jindal & Liu, 2008). Improving the identification of opinion spam, especially fake reviews, can follow two routes: using different features in combination or applying and tuning different machine learning algorithms (Crawford et al., 2015).

This article seeks to achieve different performance outcomes by using the same dataset and algorithm (J. Li et al., 2014; Ott et al., 2011). The difference that arises can be explained by the different use of the variables - that is, feature sets - by adding features to review-based features and measuring their different performance in prediction (Jindal & Liu, 2008). Researchers have often used different features from one or two feature sets in combination to identify fake reviews, as an overview of the distinct use of features and feature sets has shown (Asghar et al., 2020). In work with only one feature set, researchers have used review-based features from a hotel dataset with two supervised machine learning approaches (e.g., naive Bayes (NB) and SVM) to identify fake reviews and achieved 86% accuracy when applying SVM on the synthetic dataset (Ott et al., 2013). Others have used reviewer-based features to identify spammers with self-collected datasets from Amazon.com, and by using only a limited set of features applying SVM to detect spammers achieved an accuracy of 93% (Nair et al., 2016). Still others created a system to detect fake reviews via the sentiment analysis of product attributes and achieved 83% accuracy relative to human detection (Zhiyuli et al., 2015). In work involving two feature sets, researchers have achieved 72% accuracy with a dataset combining insights from Amazon.com and TripAdvisor.com data and using review- and reviewer-based features (Fei et al., 2013). The focus of that approach is exploiting burstiness in reviews by applying a Markov random field followed by loopy belief propagation for graph analysis. Others have applied SVM using the Yelp.com dataset and review- and reviewer-based features, also, for the outcome of 87% accuracy (Elmogy et al., 2021). Noekhah et al. used a self-created Amazon dataset to show how reviewer and review features can improve the accuracy of classification, as demonstrated in his work. An accuracy of 93% was achieved (Noekhah et al., 2014). However, combining product-, review-, and reviewer-based feature sets using machine learning algorithms has not yet been performed, possibly due to the lack of labeled datasets containing deceptive review spam and the use of synthetic datasets, which can problematically not represent real-world problems (Crawford et al., 2015). This problem is vanished in the article because of the applied data collection process by Martens & Maalej, 2019.

This research paper addresses the improvement of systems designed to detect deceptive opinion spam reviews by intensive feature engineering and, as a result, shows the benefits of using all three mentioned feature sets together. Therefore, a promising and frequently used algorithm is applied. The use of sophisticated methods, such as multi-class text classifications, using BERT, is deliberately avoided in favor of explainability and comparability.

3. Experimental Design

3.1 Dataset

Due to the lack of real-world gold-standard datasets, to experiment with the use of multiple feature sets a previously formed dataset from Martens & Maalej containing reviews of apps in the iOS Apple App Store is used. This dataset consists of two parts. The first part contains the official dataset of 62,617,037 reviews from March 2017, which was collected by crawling the App Store addressing 1,430,091 different apps with corresponding metadata and reviews. The second part contains the fake review dataset with 60,431 fake reviews. These reviews were collected in April 2017 by social investigation, crawling, and request to application programming interfaces, and received mainly as image files. The fake reviews were preprocessed and cleaned in three steps: the extraction of textual data from image files, English-language filtering, and the de-duplication of reviews. Next, the preprocessed reviews were compared with the reviews from the official review dataset, and matches were kept. After those steps, 8,607 fake reviews of 1,929 apps written by 721 reviewers remained. This is where the advantage of this data set becomes clear, namely that a fake review is really a fake review. The top three genres of the reviewed apps were games (53%), photo and video (6%), and health and fitness (4%). For further analysis, the dataset was balanced to have 50% reviews labeled as fake and 50% labeled as non-fake, namely 8,000 randomly selected deceptive opinions and 8,000 randomly selected official reviews.

With that dataset, three feature sets were created. In the following, the components of the three feature sets are described and one feature each is visualized as variable description (VD). First, the reviewer-based feature set contained the total number of reviews provided per reviewer (*user_review_count*), the percentage of reviews per star rating (*user_given_%-age_x*_reviews*), the lifetimes of the reviewers' accounts (*user_lifetime*), and the average time between all reviews provided by each reviewer (*user_frequency*). The latter feature was shown in the histogram in Figure 1. The two types of reviews - fake review and non-fake review - were distinguished by color and plotted according to their occurrence. The period up to three months is visualized. One bin represents one day. It can be recognized that the time difference between two reviews is smaller for fake reviews than for non-fake reviews. The second feature set was the product-

based feature set, which contained the total number of reviews received for all versions of each app (*product_review_count*) and the percentage of reviews received per star rating (*product_received_%-age_x*_reviews*). Figure 2 shows the frequency of 1 star rating for the products for the two types of reviews. Last, the review-based feature set contained the review's text and their lengths, as count of characters (*length_of_review*). Figure 3 displays the amount of characters of a review for the two types of reviews.

Overall, the dataset is particularly suitable for identifying deceptive opinion spam reviews because the flagged fake reviews meet the criteria of deceptive opinions and because it has a balanced distribution of fake and non-fake reviews. All three feature sets were derived from the literature and extracted from collected data from Martens & Maalej. Although the dataset is ideal for achieving the objective being certain, that an identified fake review is for sure a fake review, it cannot be ruled out that the reviews of apps classified as being correct are non-fake reviews, because those entries may not have been included in the collected dataset of fake reviews.

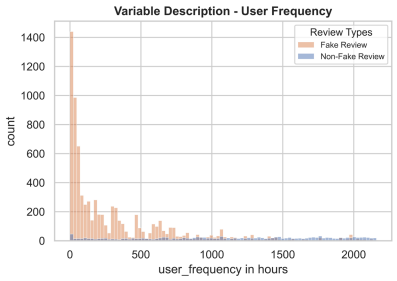

Figure 1. VD-User Frequency.

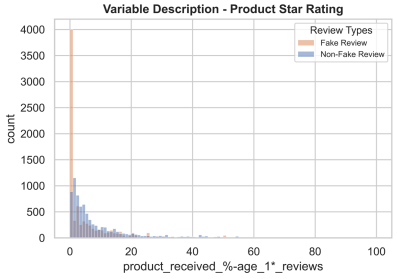

Figure 2. VD-Product Star Rating.

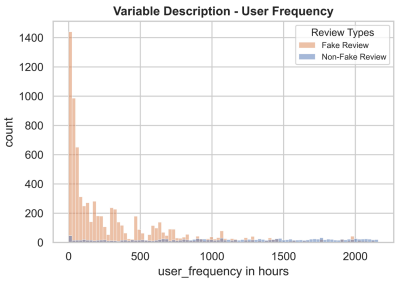

Figure 3. VD-Review Length.

3.2 Proposed Approach

The proposed approach is based on the process models CRISP-DM (Chapman et al., 2000) and ASUM-DM (IBM Corporation, 2016). Figure 4 displays the process, which depicts the approach's four steps: (A) data preprocessing, (B) feature extraction, (C) classification task, and (D) model evaluation.

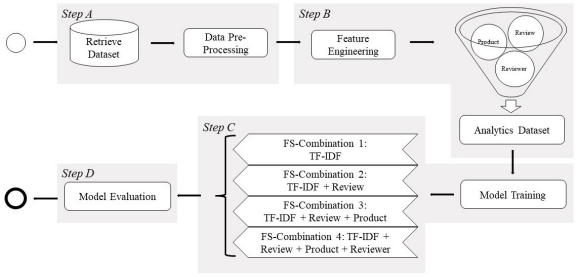

Figure 4. Process model.

First, in Step A, the dataset of reviews is loaded, and the texts of the reviews are preprocessed. During the preprocessing several manipulations are executed: the removal of HTML tags, the resolution of contractions, the removal of diacritics, the lowercasing of all words, the removal of stop words, lemmatization, and stemming with a Porter stemmer (Porter, 1980). Due to the dataset's aggregated format, it is not feasible to extend the reviewer- and product-based feature sets. Therefore, those two feature sets are used unchanged.

Second, in Step B, after the dataset is cleaned, the following review-based features are extracted. First the feature term frequency–inverse document frequency, TF-IDF, is used to reflect the relative importance of certain words in the reviews determined by weighting factor (Witten et al., 2011). To calculate the TF-IDF, n-grams are used. These are contiguous sequence of n words from a given sample of text. Each review is split into sequences of three words, resp. trigrams, calculated with the preprocessed texts of the reviews displaying the value of their relative importance. The term frequency (TF) defines the relative frequency of a term, resp. trigram, within a review. The inverse document frequency (IDF) is a measure of a term's rarity across all reviews, indicating term-specificity. A multiplication of the two statistics (1) and (2) provides TF-IDF as given by Eq. (3).

$$TF(t,r) = \frac{f_{t,r}}{|\{f_{t',r} : t' \in r\}|} \quad (1)$$

In Eq. (1) $f_{t,r}$ indicates the frequency of a term t occurring in a review r. $f_{t',r}$ presents the frequency of any term t' such that $|\{f_{t',r} : t' \in r\}|$ displays the overall number of terms in reviews r.

$$IDF(t,R) = log\left(\frac{|R|}{|\{r' \in R : t \in r'\}|}\right) \quad (2)$$

R is the review corpus and $|R|$ indicates the overall number of reviews in the analysed review dataset, resp. corpus. The term $t, |\{r' \in R : t \in r'\}$ provides the overall number of documents in the review dataset R that contain term t.

$$TF\text{-}IDF(t,r,R) = TF(t,r) * IDF(t,R) \quad (3)$$

Average sentence length (*avg_sentence_length*) can be calculated as:

$$avg(sentence\ length) = \frac{count\ of\ words}{count\ of\ sentence} \quad (4)$$

Next, the average word length (*avg_word_length*) can be calculated as:

$$avg(word\ length) = \frac{count\ of\ letter}{count\ of\ words} \quad (5)$$

After that, the number of sentences (*count_sentences_review*) using a published procedure by (Kiss & Strunk, 2006) is computed. Once the sentiment is calculated by using the TextBlob Naive Bayes Analyzer (Loria, 2018), values for the positive and negative sentiments are defined for each sentence in the review. That information is aggregated to the review by calculating the metrics of minimum value, first/ second/ third quartile, and maximum value for the review's sentiment (*sentiment_[pos/neg]_Qx*). The review-based feature set is enriched by the mentioned 13 features and the TF-IDF data.

In Step C, the model is trained. To test the hypothesis that using all three feature sets improves the classification performance of the algorithms, the random forest method is used for two reasons. On the one hand, it is the best-performing method for the classification task to identify fake reviews in (Martens & Maalej, 2019); on the other, its explainability and applicability (Breiman, 2001). The method uses the three defined feature sets as input features in four feature set combinations (FS-Combination). The data are input as compressed sparse row matrices. The four combinations are used as input with different compositions of features, defined in Table 1:

Table 1. Feature set combinations.

FS-Combination 1	TF-IDF
FS-Combination 2	TF-IDF + further review features from Step B (review features)
FS-Combination 3	TF-IDF + review features + product features
FS-Combination 4	TF-IDF + review features + product features + reviewer features

For each FS-Combination a computation of the random forest is done. In this context the Bayes Search Cross Validation (BSCV) for the optimal hyperparameter set is conducted. Therefore, a maximum of 100 runs, depending on a callback, defined as difference in the F1 score regarding the last five executed runs, is executed. If the difference in the score is less than 0.01 then the process stops and outputs the optimal parameters (Pedregosa et al., 2011). The used search space is defined in Table 2:

Table 2. BSCV parameter spaces.

Number of trees in the forest	[100, …, 5000]
The number of features to consider when looking for the best split	[auto, sqrt, log2]
The function to measure the quality of a split	[gini, entropy]
The maximum depth of the tree	[6, …, 110]
The minimum number of samples required to split an internal node	[2, …, 100]
The minimum number of samples required to be a leaf node	[2, …, 10]
Usage of bootstrap samples when building trees	[True, False]

A 10-fold cross-validation (Hastie et al., 2009) is applied during the prediction to improve the robustness and generalizability of the algorithm. Therefore k-1 subsets are used to train the data and the last one is left out for testing. Afterwards the model is averaged against each of the folds.

Step D is used to statistically evaluate the results in three steps. First, the quality of the algorithm is determined by the metrics of accuracy, recall, F1 score, and area under the Receiver Operating Characteristics (AUC) (Idrees et al., 2017; Pedregosa et al., 2011). In the binary classification problem, the definitions are shown in Eq. (6)-(9). Further the graphical representation of the AUC as Receiver Operation Characteristics (ROC) is used to plot the trade-off.

$$Accuracy = \frac{tp + tn}{tp + fn + fp + tn} \quad (6)$$

$$Recall = \frac{tp}{tp + fn} \quad (7)$$

$$F1 = \frac{\left(2 * \frac{tp}{tp + fp} * \frac{tp}{tp + fn}\right)}{\left(\frac{tp}{tp + fp} + \frac{tp}{tp + fn}\right)} \quad (8)$$

$$AUC = \frac{1}{2}\left(\frac{tp}{tp + fn} + \frac{tn}{tn + fp}\right) \quad (9)$$

tp, *fp*, *fn* and *tn* are true positive, false positive, false negative, and true negative classifications of the reviews, respectively. A threshold of 0.5, as proposed by the method, is used, generating the predictions (Pedregosa et al., 2011). Accuracy is defined as the ratio of correct predictions to total predictions made. Recall is the ratio of instances of reviews predicted to be fake among all the instances that are fake; it indicates the strength of a classifier in detecting deceptive opinion spam. F1 combines both precision, as ratio of correct positive predictions to the total predicted positive predictions, and recall by taking their weighted harmonic mean, because improving precision and recall at simultaneously can be conflicting, since simple averaging does not punish extreme values (Rastogi et al., 2020).

To further validate the results, besides the evaluation metrics, a graphical analysis using ROC is

performed. It shows the relative trade-off between true positive rate and false positive rate at different discriminating thresholds and therefore provides a better understanding of a classifier's performance (Fawcett, 2006). A single metric, AUC, is used to summarize the results of the ROC by providing an aggregate measure of performance (Fawcett, 2006).

To avoid the exclusive effect of the use of the feature sets on the performance of the classification, feature importance is considered in the context of the implementation of the random forest. The Gini impurity, as measure of non-homogeneity (Breiman et al., 1998), is used, as defined in Eq. (10)

$$Gini\ Index = \sum_i p_i(1-p_i)\ with\ p_1(1-p_1) + p_2(1-p_2) \quad (10)$$

p_i is the probability of class i, and the interval of Gini is [0,0.5]. For the two-class problem used, the Gini impurity for a given node is given as shown above. A pure node is defined by a probability of 0, and the largest Gini score is 0.5, in which case the purity of the node is the smallest. Therefore, when training a tree, how much each feature contributes to decreasing the weighted impurity is computed. By applying the random forest algorithm, the decrease in impurity over trees is averaged.

The proposed approach is modelled in Python 3.8 primarily using the packages nltk, pandas, scipy, scikit-learn, spacy, and textblob.

4 Results and Discussion

In the following, the results are shown first. Table 3 presents the accuracy, recall, F1, and AUC values for the four FS-Combinations. Figure 5 allows a comparison of the three first mentioned results, visualized graphically as line plots.

The BSCV results in the following parameter setting: 5000 classification trees, maximum amount of $log2\ (x)$ number of features, 'gini' as split criterion, a maximum tree depth of 6, a minimum number of samples to split in a node of 50, a minimum number of samples to be a leaf node of 2 and no bootstrapping.

Table 3. Evaluation metrics.

	Evaluation metrics			
Used FS-Combination	**Accuracy**	**Recall**	**F1**	**AUC**
FS-Combination 1	0.7917	0.7411	0.7806	0.7917
FS-Combination 2	0.8092	0.8663	0.8195	0.8092
FS-Combination 3	0.9216	0.9209	0.9215	0.9216
FS-Combination 4	0.9438	0.9430	0.9437	0.9438

A clear added value can be achieved by using at least two feature sets due to the slight increase of the F1 score (0.0389) by 4.98%, with respect to the achieved difference between FS-Combination 1 and FS-Combination 2. That difference can be explained by the features used. The TF-IDF data is enriched with additional features from the same feature set. The improved prediction achieved by using another feature set manifested in the difference of the F1 score of 12.45% (0.1020) from FS-Combination 2 to FS-Combination 3, which achieved an increase of 249.76% compared with the previous difference. To support the addition of a further feature set, the FS-Combination (*TF-IDF + review + product*) is calculated using the reviewer-based feature set instead of the product-based feature set. Here, the result deviates by only 0.1% in comparison to the accuracy of FS-Combination 3 and is therefore not listed as an extra FS-Combination. Nevertheless, this is a clear indication that this feature set makes an important contribution to the identification of fake reviews and thus motivates the use of the reviewer-based features in FS-Combination 4. Another increase occurs from FS-Combination 3 to FS-Combination 4, in which the third feature set is also used for prediction. Although relevant, the increase (2.41%) is less than the increase with the first addition of a feature set. By including all three feature sets, the model's F1 score increases from 0.7806 to 0.9437 (20.89%). That result is supported by an increase of 0.1521 (19.21%) in accuracy and by an increase of 0.2019 (27.24%) in recall. These changes are visualized in Figure 5 using line plots displaying that the strong improvement of the three metrics was achieved by using three instead of two, as usually used feature sets.

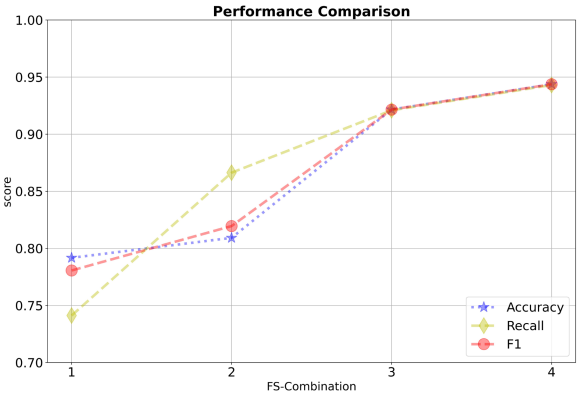

Figure 5. Performance Comparison.

Figure 6 presents the improvement achieved by using the different feature sets, which is also apparent in the AUC values shown in Table 3. In the figure, the curve of the fourth FS-Combination is the closest to the top-left corner, which indicates its superior performance. The first FS-Combination, meanwhile, is shown to have the least accurate performance. To show the results in Figure 6 in single metrics, the AUC is

presented; the AUC score increases by 19.21% from the first to the fourth FS-Combination.

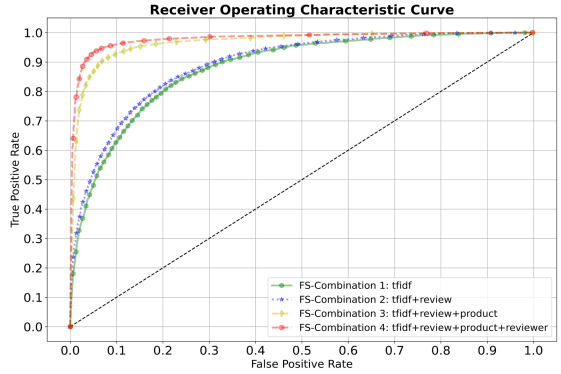

Figure 6. Receiver Operating Characteristics Curve.

Figure 7 shows the importance of the top 20 input features of the random forest, determined by using the fourth FS-Combination, with the three feature sets displayed in different colors. Nine review-based, seven reviewer-based, and four product-based variables are presented. The most important variable is the percentage of one-star ratings (*product_received_%-age_1*_reviews*) from the product-based feature set to classify the data, followed by a reviewer-based feature (*user_frequency*) and a review-based feature (*length_of_review*). In the case of the strongest feature can be concluded, that a high percentage of one star ratings of a product tends to be a good feature to decrease the node impurity and also tend to be one of the first splits in a random forest. A TF-IDF feature also included the trigram *str('graphically stunning game')*. The top 20 variables account for 22.37% of all feature importance and $9.42 * 10^{-5}$% of the features used. In sum, the displayed review-based features defined 9.50% of the feature importance, followed by reviewer-based features (7.95%) and product-based features (4.91%). The problem of inflating the importance of continuous features or categorical variables with high cardinality by using feature importance does not exist due to the design of the dataset.

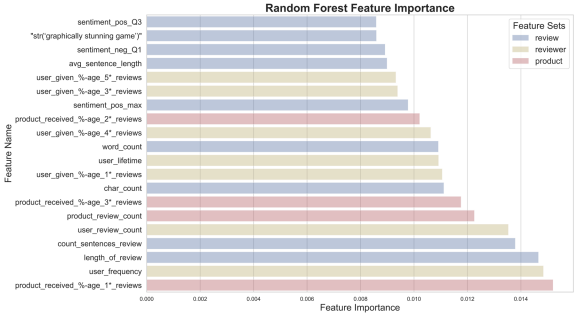

Figure 7. Feature Importance.

Taking a closer look at the results, it is noticeable that product stars related features are disproportionately represented, with 7/20 features. It is referred three times to the received stars of the product from the product-based feature set (*product_received_%-age_[1,2,3]*_reviews*) and four times to the given stars from reviewer point of view (*user_given_%-age_[1,3,4,5]*_reviews*) - reviewer-based feature set. That the variables contribute a high proportion to the separation between fake reviews and non-fake reviews is already evident from Figure 2. In this figure, a clear peak in the fake review distribution can be seen - that products which receive fake reviews receive very few 1-star ratings in percentage terms. This can be explained by the primary intention of creating fake reviews, which is to positively influence products in order to gain a competitive advantage over other products in various forms.

The relevance of the listed features from the reviewer-based feature set, such as *user_frequency*, *user_review_count* and *user_lifetime*, could depend on the used data sources. Most of the fake reviews were collected from databases of fake review portals. This explains why the reviews of private individuals differ significantly from those of people who earn money with them in terms of frequency and account lifetime. To earn money with fake reviews, several fake reviews are typically written in a short period of time. In the analysis of the corresponding feature, fake reviewers post reviews four times more frequently than normal reviewers. This is evident from an analysis of the data presented in Figure 1, which shows a significantly higher frequency for the creation of fake reviews compared to non-fake reviews. This can be used to explain the first two features. The lifetime of a fake reviewer account is significantly longer than that of a private individual. This indicates that they are not recognized by the operator of the platform used and are not deleted accordingly.

With respect to the review-based features, the length of the reviews is specified with the features *lenght_of_review*, *count_sentences_review*, *char_count* and *word_count*, among others. Figure 3 shows the difference in the length of reviews. In the deeper analysis of the numerical information of the reviews, it is noticeable that the average review length of the reviews does not vary much between fake reviews and non-fake reviews, but the median number of characters for fake reviews includes almost 50% more characters than non-fake reviews. A significant difference in the median is also evident when differentiating based on the number of words. The difference in the average values is inconspicuous. The fact that fake reviews are longer than non-fake reviews does not correspond to the first expectation but is understandable regarding the data

collection source which pinpoints that people are paid for creating the fake reviews and that these should be correspondingly difficult to identify.

It can be shown in two ways that a relevant improvement in the performance of the identification of deceptive opinion spam can be achieved by using multiple feature sets. Furthermore, the results of the feature importance indicate, that features from all three feature sets have high feature importance. Therefore, it can be concluded that the combination of variables out of the three feature sets improves the fake review detection performance of the algorithm. This does not support the general proposition that more variables are always better for prediction, since the Random Forest uses the strongest parameter selection constraint ($log2$) for a corresponding split.

It can thus be concluded that for detecting fake reviews, using a wide range of data, resp. different feature sets, can improve the performance of prediction, as can be supported via comparison with published work (e.g., Elmogy et al., 2021; Noekhah et al., 2014) in which better accuracy was also achieved by using multiple feature sets.

In the work of Emolgy et al. the application of several machine learning algorithms on different feature sets were used to analyze the change in the identification of fake reviews on the Yelp.com dataset. Review-based features were used as a basis, which were supplemented by reviewer-based features. By using the review-based features alone, an F1 score of 0.8230 was achieved. By adding the second feature set a F1 score of 0.8373, an increase of 3.80%, was achieved. In the present paper, an increase of 12.45% in the F1 score can be achieved by using two feature sets (FS-Combination 3). If the third feature set is also used, then an increase in F1 score just under five and a half times (549.85%) compared to the result of Elmogy et al. has been achieved. Noekhah et al. show in the paper using a self-created Amazon dataset, using a graph-based model, that an improvement in accuracy can be achieved by combining feature sets. Basically, review and reviewer features are extracted from the data. Using mainly reviewer variables, an accuracy of 0.79 is achieved. When adding selected review variables, the accuracy can be increased by 3.80% (0.82). Subsequently, all available variables were used, resulting in an overall increase of 0.14 (0.93). This is a percentage increase of 17.72%. In this study, the added value of adding another feature set is not immediately apparent. The improvement is substantiated by adding all available features. Compared to the improvement obtained by Noekhah et al. an improvement of 20.89% (ca. 17.79% higher) is obtained by the performed application of three feature sets in this presented paper.

A limitation of the dataset - namely, that an increased number of false-negative predictions may remain undetected - cannot be excluded from the design of data collection process. In order to substantiate the results, it would be useful to transfer the findings to other domains, as more data sets become available, to determine whether a similar improvement can be achieved by using different sources, resp. domains of the data.

5 Conclusion

By applying a four-step process, it was shown that the use of different feature sets (review-, reviewer-, and product-based) can add significant value to the classification of fake reviews than solely considering textual features based on the TF-IDF, as often used in the literature. Labeled reviews from the iOS App Store in combination with two statistical approaches are used to demonstrate that the simultaneous utilization of multiple feature sets enhances the detection of fake review. First, by using enriched feature set combinations as input for the random forests, it was found that the performance of classification increased with the number of feature sets used. By including three feature sets (e.g. reviewer- and product-based) instead of the review-based feature set alone, an increase in the harmonic mean of precision and recall was achieved by 20.89%. With an increase of 19.21%, the AUC displayed similar information. Second, results for feature importance indicate that none of the feature sets can be neglected, for considering the top 20 features, variables from all three sets were represented. Thus, it can be concluded that a comprehensive examination of a review and data about the product and the reviewer is essential for identifying fake reviews. At least two different feature sets should be used to increase the effectiveness of fake review detection algorithms. This research paper provides the first numerical assessment of the added value of using the feature sets most commonly used in the literature for identifying fake reviews. In this context, a relevant increase in classification performance is achieved, which could yield economic significance. The costs incurred using a broader FS-Combination could be compensated by a substantial increase in classification performance.

For that reason, owners of online stores or portals can identify fake reviews more easily than people who would have to collect the data themselves (e.g., with web scraping), for it is far more cumbersome to aggregate correlations according to users' movement and product data. It is conceivable to link review-related information with customers' journeys in order to better assess the deceptive opinion spam. The procedure

applied can be performed by reducing the input feature or by applying a principal component analysis in a more concrete form to strengthen the results of feature importance.

Further, to strengthen the fact, different approaches to reduce the applied number of features as Shapley value or local interpretable model-agnostic explanations could be applied. As another direction for future research, it can be deduced from this article that feature engineering should play an important role in improving the performance of identifying fake reviews. The results could also be supported by applying neural network approaches and using additional data, including the transaction data of customers on online portals, to model other relevant factors. The influence of social media companies could also come into focus. Here, groups for the creation of fake reviews could be looked at more closely and a connection made between the publicly available information and published reviews. Last, it is proven that the usage of different feature sets increases the performance, ongoing it can be investigated which algorithms are most suitable for the identification of fake reviews using a broad range of data.

References

Asghar, M. Z., Ullah, A., Ahmad, S. & Khan, A. (2020). Opinion spam detection framework using hybrid classification scheme. Soft Computing, 24(5), 3475–3498.

Askalidis, G. & Malthouse, E. C. (2016). The Value of Online Customer Reviews. In Proceedings of the 10th ACM Conference on Recommender Systems, 155–158.

Aslam, U., Jayabalan, M., Aziz, H. I. & Sohail, A. (2019). A Survey on Opinion Spam Detection Methods. In International Journal of Scientific & Technology Research, 8, 1355–1363.

Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5–32.

Breiman, L., Friedman, J., Stone, C. J. & Olshen, R. A. (1998). Classification and regression trees (1. CRC Press repr). Chapman & Hall/CRC.

CHEQ (2021). Fake Online Reviews 2021: The Economic Cost of Bad Actors on the Internet. https://f.hubspotusercontent00.net/hubfs/5228455/Research/Fake%20Online%20Reviews%202021.pdf.

Crawford, M., Khoshgoftaar, T. M., Prusa, J. D., Richter, A. N. & Al Najada, H. (2015). Survey of review spam detection using machine learning techniques. In Journal of Big Data, 2(1).

Dellarocas, C. (2006). Strategic Manipulation of Internet Opinion Forums: Implications for Consumers and Firms. In Management Science, 52(10), 1577–1593. https://EconPapers.repec.org/RePEc:inm:ormnsc:v:52:y:2006:i:10:p:1577-1593

Dong, L., Ji, S., Zhang, C., Zhang, Q., Chiu, D., Qiu, L. & Da Li (2018). An unsupervised topic-sentiment joint probabilistic model for detecting deceptive reviews. In Expert Systems with Applications, 114, 210–223.

Elmogy, A., Tariq, U., Mohammed, A. & Ibrahim, A. (2021). Fake Reviews Detection using Supervised Machine Learning. International. In Journal of Advanced Computer Science and Applications, 12.

Engström, P. & Forsell, E. (2018). Demand effects of consumers' stated and revealed preferences. In Journal of Economic Behavior & Organization, 150, 43–61.

Fawcett, T. (2006). An introduction to ROC analysis. In Pattern Recognition Letters, 27(8), 861–874.

Fei, G., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M. & Ghosh, R. (2013). Exploiting Burstiness in Reviews for Review Spammer Detection. In Proceedings of the International AAAI Conference on Web and Social Media, 7(1), 175–184.

Filieri, R., Alguezaui, S. & McLeay, F. (2015). Why do travelers trust TripAdvisor? Antecedents of trust towards consumer-generated media and its influence on recommendation adoption and word of mouth. In Tourism Management, 51(C), 174–185.

Gu, B., Park, J. & Konana, P. (2012). Research Note —The Impact of External Word-of-Mouth Sources on Retailer Sales of High-Involvement Products. In Information Systems Research, 23(1), 182–196.

Hastie, T., Tibshirani, R. & Friedman, J. (2009). The Elements of Statistical Learning. Springer New York.

He, S., Hollenbeck, B. & Proserpio, D. (2020). The Market for Fake Reviews. In SSRN Electronic Journal.

Hunt, K. M. (2015). Gaming the system: Fake online reviews v. consumer law. In Computer Law & Security Review, 31(1).

IBM Corporation. (2016). Analytics Solutions Unified Method: Implementations with Agile principles. ftp://ftp.software.ibm.com/software/data/sw-library/services/ASUM.pdf

Idrees, F., Rajarajan, M., Conti, M., Chen, T. M. & Rahulamathavan, Y. (2017). PIndroid: A novel Android malware detection system using ensemble learning methods. In Computers & Security, 68, 36–46.

Jiménez, F. R. & Mendoza, N. A. (2013). Too Popular to Ignore: The Influence of Online Reviews on Purchase Intentions of Search and Experience Products. In Journal of Interactive Marketing, 27(3), 226–235. h

Jindal, N. & Liu, B. (2008). Opinion Spam and Analysis. In WSDM '08, Proceedings of the 2008 International Conference on Web Search and Data Mining, 219–230.

Kamalesh, M. D. & Diwedi, H. K. (2015). Extracting product features from consumer reviews and its applications. In International Journal of Applied Engineering Research, 10, 2345–2350.

Kiss, T. & Strunk, J. (2006). Unsupervised Multilingual Sentence Boundary Detection. In Computational Linguistics, 32(4),

Lau, R. Y. K., Liao, S. Y., Kwok, R. C.-W., Xu, K., Xia, Y. & Li, Y. (2012). Text Mining and Probabilistic Language Modeling for Online Review Spam Detection. In Transactions on Management Information Systems, 2(4).

Li, J., Ott, M., Cardie, C. & Hovy, E. (2014). Towards a General Rule for Identifying Deceptive Opinion Spam. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics , 1, 1566-1576.

Li, X. & Hitt, L. M. (2010). Price Effects in Online Product Reviews: An Analytical Model and Empirical Analysis. In MIS Quarterly, 34(4).

Loria, S. (2018). TextBlob: Simplified Text Processing. Release 0.15, 2.

Luca, M. & Zervas, G. (2016). Fake It Till You Make It: Reputation, Competition, and Yelp Review Fraud. In Management Science, 62(12), 3412–3427.

Luyang, L., Bing, Q., Wenjing, R. & Ting, L. (2017). Document representation and feature combination for deceptive spam review detection. In Neurocomputing, 254, 33–41.

Martens, D. & Maalej, W. (2019). Towards understanding and detecting fake reviews in app stores. In Empirical Software Engineering, 24(6), 3316–3355.

Nair, A., Phapale, A., Yagnik, V. & Bathe, K. (2016). Opinion Spam Mining. In International Research Journal of Engineering and Technology, 3(4), 1855–1859. https://www.irjet.net/archives/V3/i4/IRJET-V3I4366.pdf

Naveed, H., Hamid, T. M., Ghulam, R., Ibrar, H. & Mohammad, K. (2019). Spam Review Detection Techniques: A Systematic Literature Review. In Applied Sciences, 9(5), 987.

Noekhah, S., Fouladfar, E., Naomie, S. & Ghorashi, S. H. (2014). A Novel Approach for Opinion Spam Detectionin E-Commerce. In Proceedings of the 8th IEEE international conference on E-Commerce with focus on E-trust.

Ott, M., Cardie, C. & Hancock, J. (2013). Negative Deceptive Opinion Spam, 497–501.

Ott, M., Choi, Y., Cardie, C. & Hancock, J. (2011). Finding Deceptive Opinion Spam by Any Stretch of the Imagination.

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

Porter, M. F. (1980). An algorithm for suffix stripping. In Program, 14(3).

Rastogi, A., Mehrotra, M. & Ali, S. S. (2020). Effective Opinion Spam Detection: A Study on Review Metadata Versus Content. In Journal of Data and Information Science, 5(2), 76–110.

Ren, Y. & Ji, D. (2017). Neural networks for deceptive opinion spam detection: An empirical study. In Information Sciences, 385-386, 213–224.

Ren, Y. & Ji, D. (2019). Learning to Detect Deceptive Opinion Spam: A Survey. In IEEE Access, 7, 42934–42945.

Ren, Y., Ji, D. & Zhang, H. (2014). Positive Unlabeled Learning for Deceptive Reviews Detection. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 488–498.

Salehi-Esfahani, S. & Ozturk, A. (2018). Negative reviews: Formation, spread, and halt of opportunistic behavior. In International Journal of Hospitality Management.

Schuckert, M., Liu, X. & Law, R. (2016). Insights into Suspicious Online Ratings: Direct Evidence from TripAdvisor. In Asia Pacific Journal of Tourism Research, 21(3), 259–272.

Simon-Kucher (2019). The Rating Economy: Consumer Survey. Bonn. https://www.simon-kucher.com/en/TheRatingEconomy-Overview.

Song, W., Park, S. & Ryu, D. (2017). Information Quality of Online Reviews in the Presence of Potentially Fake Reviews. In Korean Economic Review, 33, 5–34.

Sun, C., Du, Q., Tian, G. & Chen, C. (2016). Exploiting Product Related Review Features for Fake Review Detection. In Mathematical Problems in Engineering, 2016.

Witten, I. A., Eibe, F. & Hall M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques.

Wu, Y., Ngai, E. W., Wu, P. & Wu, C. (2020). Fake online reviews: Literature review, synthesis, and directions for future research. In Decision Support Systems, 132.

Yu, C., Zuo, Y., Feng, B., An, L. & Chen, B. (2019). An individual-group-merchant relation model for identifying fake online reviews: an empirical study on a Chinese e-commerce platform. In Information Technology and Management, 20(3), 123–138.

Zhiyuli, A., Liang, X. & Wang, Y. (2015). Discerning the Trend: Concealing Deceptive Reviews. In 2015 IEEE International Conference on Systems, Man, and Cybernetics, 1833–1838.