International Journal of Uncertainty,
Fuzziness and Knowledge-Based Systems
Vol. 25, Suppl. 2 (December 2017) 151–174

DOI: 10.1142/S0218488517400165

Detecting Deceptive Opinions: Intra and Cross-Domain Classification Using an Efficient Representation

Leticia C. Cagnina

CONICET (Argentina), LIDIC, Universidad Nacional de San Luis,
Ejército de los Andes 950, San Luis, Argentina
lcagnina@unsl.edu.ar

Paolo Rosso

PRHLT Research Center, Universitat Politècnica de València,
Camino de Vera s/n, Valencia, España,
prosso@dsic.upv.es

Received 15 November 2016
Revised 29 June 2017

Online opinions play an important role for customers and companies because of the increasing use they do to make purchase and business decisions. A consequence of that is the growing tendency to post fake reviews in order to change purchase decisions and opinions about products and services. Therefore, it is really important to filter out deceptive comments from the retrieved opinions. In this paper we propose the character n-grams in tokens, an efficient and effective variant of the traditional character n-grams model, which we use to obtain a low dimensionality representation of opinions. A Support Vector Machines classifier was used to evaluate our proposal on available corpora with reviews of hotels, doctors and restaurants. In order to study the performance of our model, we make experiments with intra and cross-domain cases. The aim of the latter experiment is to evaluate our approach in a realistic cross-domain scenario where deceptive opinions are available in a domain but not in another one. After comparing our method with state-of-the-art ones we may conclude that using character n-grams in tokens allows to obtain competitive results with a low dimensionality representation.

Keywords: Cross-domain evaluation; deception detection; intra-domain evaluation; low dimensionality representation; opinion spam.

1. Introduction

With the increasing availability of review sites, blogs and recommendation systems, consumers rely more than ever on online reviews to make their purchase decisions. Spam is commonly present on the Web through of fake opinions or malicious comments posted in electronic commerce sites and blogs. The purpose of these kinds of spam is to promote products and services, or simply damage their reputation.

A *deceptive* opinion spam can be defined as a fictitious opinion written with the intention to sound authentic in order to mislead the reader. An opinion spam usually is a short text written by an unknown author using a not very well defined style. These characteristics make the problem of automatic detection of opinion spam very challenging. A recent survey[a] found that 74% of consumers have reinforced the decision to purchase a product or service reading positive online reviews, 60% of consumers have rejected a business after reading negative reviews, 91% of consumers read the online reviews to judge a local business or a product (68% of them form an opinion reading 1–6 reviews) and 84% of people trust online reviews as much as a personal recommendation. Therefore, detecting deceptive opinions among all retrieved ones, is a very important task.

In this paper we study the feasibility of using n-grams in tokens together with other features for the detection of deceptive opinions. We also investigate if considering also the sentiments information of a review may help. Moreover, information about the usage of pronouns, articles and verbs (in present, past and future) was also taken into account. Previous works have shown some evidence regarding the word categories more implicated in deception, that is, the use of certain pronouns and articles, words related to emotions and motion verbs.[11,14,29]

We evaluated the proposed features with a Support Vector Machines (SVM) classifier considering two experiments: intra-domain and cross-domain classifications. For intra-domain experiments we use a corpus of 1600 reviews of hotels.[30,31] We show an experimental study evaluating single features and combining them. We finally selected the best combination and compared it with state-of-the-art methods. The obtained results show that the proposed features can capture information both from the content of the reviews and their writing style, allowing to obtain classification results as good as the ones obtained by the best methods but with a lower dimensionality representation.

Then, we considered the realistic cross-domain experimental scenario where deceptive opinions are available in one domain and we need to carry out the evaluation in another one. For that we used a multi-domain corpus[22] which includes reviews of hotels, doctors and restaurants. We performed the cross-domain classifications showing that using our low dimensionality representation we can obtain acceptable results.

Finally, we may conclude that the proposed low dimensionality representation seems a good alternative for deceptive opinions detection in both intra and cross-domain scenarios.

The rest of the paper is organized as follows. Section 2 briefly describes previous works on deceptive opinions detection. Section 3 introduces the proposed features. Section 4 describes the single and cross-domain corpora used in the experiments. Section 5 illustrates the experimental study performed; first, the selection of

[a]Local Consumer Review Survey 2016 (visited: May 2, 2017): https://www.brightlocal.com/learn/local-consumer-review-survey/

features and classification for intra-domain study is shown, then the cross-domain classification and the comparison of results. Finally, in Section 6 we draw some conclusions and discuss future work.

2. Related Work

The first works for detecting fake opinions mainly considered unsupervised approaches trying to identify duplicate content,[18,23] and searching for unusual review patterns[19] or simply groups of opinion spammers.[28] More elaborate approaches included the construction of heterogenous graphs with reviewers, reviews and stores.[38] Then, making an iterative computation on the interactions between the nodes in the graph, the approach can identify suspicious reviewers, fake reviews and non reliable stores. Following works started approaching the problem of the detection of deceptive opinions in a supervised way. Ott *et al.*[31] used the 80 dimensions of LIWC2007,[32] unigrams and bigrams as set of features with a SVM classifier. The same research line was showed in several works.[8,9,22] Li *et al.*[22] studied LIWC, POS and unigrams features, obtaining the best accuracy value with a SVM classifier and unigrams for representing the reviews. In other works,[8,9] the authors employed n-grams together with syntactic production rules derived from probabilistic context free grammar parse trees. Feng *et al.*[10] proposed profile alignment compatibility features combined with unigrams, bigrams and syntactic production rules for representing the opinion spam corpus, while Li *et al.*[21] used a generative Latent Dirichlet Allocation version based on a mixture of topics to model the reviews. The results of a logistic regression model were presented based on 13 different independent features for the representation of the reviews:[1] complexity, reading difficulty, adjectives, articles, nouns, prepositions, adverbs, verbs, pronouns, personal pronouns, positive cues, perceptual words and future tense. Then, the authors concluded that only articles and pronouns (over the 13 features) could significantly distinguish true from false reviews. More recently, a partially supervised technique named PU-learning[24] has been successfully used in text classification. Other PU-learning-based methods[16,17,34] were applied in order to learn from positive examples and unlabeled ones to detect opinion spam, using only few examples of deceptive opinions and a set of unlabeled data. Particularly, a semi-supervised model called mixing population and individual property PU-learning,[34] was presented. The model was incorporated to a SVM classifier for detecting deceptive reviews. The authors concluded that the good performance of their proposal is due to the topic information captured by the model combined with the examples and their similarities. Some PU-learning variants using two different representations: word n-grams and character n-grams[16,17] were also proposed. The best results were obtained with a Naïve Bayes classifier using character 4 and 5 grams as features[17] and, the conjunction of word unigrams and bigrams.[16] With those results the authors concluded that PU-learning showed to be appropriate for detecting opinion spam.

3. Feature Selection for Deceptive Opinions

In this section, we describe the three different kinds of features studied in this work and the tools used for their extraction.

3.1. *Character n-grams in tokens*

The main difference of character n-grams in tokens[b] with respect to the traditional NLP feature character n-grams is the consideration of the tokens for the extraction of the n-grams. That is, tokens with less than n characters are not considered in the process of extraction neither blank spaces. Character n-grams in tokens preserve the main characteristics of the standard character n-grams:[37] *effectiveness* for quantifying the writing style used in a text,[20,35] the *independence* of language and domains,[41] the *robustness* to noise present in the text,[7] and, *easiness* of extraction in any text. But unlike the traditional character n-grams, the proposed feature obtains a smaller set of attributes, that is, character n-grams in tokens avoids the need of feature dimension reduction. Figure 1 illustrates that difference.

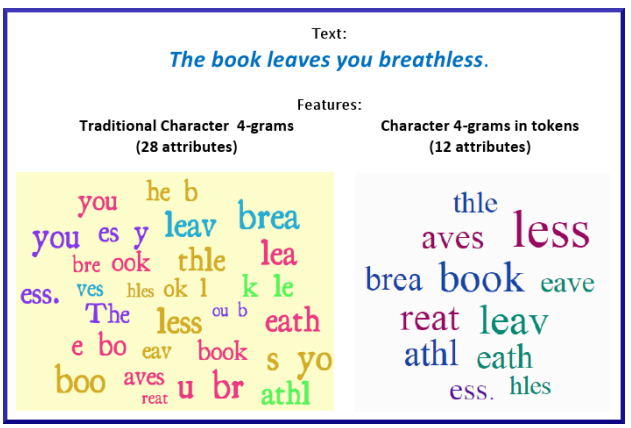

Fig. 1. Set of features obtained with traditional character n-grams and character n-grams in tokens, considering n = 4.

As it can be observed from Fig. 1 the amount of features obtained with the character n-grams in tokens is considerably less, although the effectiveness of this representation still being good, as we will see in Section 5.

For the extraction of character n-grams in tokens we have used the Natural Language Toolkit (NLTK) package[2] with Python language.

[b]Token is considered in this work as any sequence of consecutive characters separated by one or more blank spaces.

3.2. *Sentiment-based features*

Previous works have been demonstrated that the use of sentiment information helps to discriminate truthful from deceptive text.[3,14,29] There is some evidence that liars use more negative sentiments than truth-tellers. Based on that, we obtained the percentages of positive, negative and neutral sentiments contained in the sentences of a document. Then, we have used these values as features in order to represent the polarity of the text.

For the calculation of the percentages of positive, negative and neutral sentiments contained in the text we have used the Natural Language Sentiment Analysis API[c] which analyses the sentiments, labeling a text with its polarity (positive, negative or neutral). We have obtained the polarities of each sentence and then we have obtained the percentages of the polarities associated to the whole document (a review in our case). Finally, we have used those as features.

3.3. *LIWC-based features*

Several features derived from *Linguistic Inquiry and Word Count* (LIWC) were considered. In particular, we have studied those related to functional aspects of the text such as word count, adverbs, pronouns, etc. After performing an early experimental study considering the 26 different elements of the linguistic processes category in LIWC2007 as features, we have concluded that pronouns, articles and verbs (present, past and future tense) may help to distinguish fake from true reviews.

4. Data Collections

Next, we describe the first publicly available opinion spam corpus gathered and proposed for intra domain experimentation.[30,31] Then, we describe the unique (as far as we know) cross-domain gold standard corpus.[22]

4.1. *The corpus used for intra-domain experiments*

The Opinion Spam corpus[30,31] is composed of 1600 *positive* and *negative* opinions for hotels with the corresponding gold standard. From the 800 *positive* reviews,[31] the 400 truthful where mined from TripAdvisor 5-star reviews about the 20 most popular hotels in the Chicago area. All reviews were written in English, have at least 150 characters and correspond to users who had posted opinions previously on TripAdvisor (non first-time authors). The 400 deceptive opinions correspond to the same 20 hotels and were gathered using Amazon Mechanical Turk crowdsourcing service. From the 800 *negative* reviews,[30] the 400 truthful where mined from TripAdvisor, Expedia, Hotels.com, Orbitz, Priceline and Yelp. The reviews are 1

[c]http://text-processing.com/demo/sentiment/

or 2-star category and are about the same 20 hotels in Chicago. The 400 deceptive reviews correspond to the same 20 hotels and were obtained using Amazon Mechanical Turk. The left side of Table 1 summarizes the amount of reviews contained in each category (Opinion Spam corpus).

4.2. *The corpus used for cross-domain experiments*

The corpus used for cross-domain experimentation was named Deception_dataset and includes reviews of three different domains: hotels, doctors and restaurants.[22] The gold standard originally contains reviews obtained from Amazon Mechanical Turk, opinions from experts in each domain such as those of employees, and reviews obtained from customers considered as the truthful ones. Only the Hotel and Restaurant domains include positive and negative reviews. The reviews of hotels are the same ones of the Opinion Spam corpus. The reviews of restaurants correspond to the 10 most popular restaurants in Chicago and for the Doctor domain the authors collected the opinions related to 15 different doctors. Considering the three types of the sources used to obtain the reviews, the amount of documents in each domain are for Hotel: 400 positive and 400 negative reviews from turkers, 140 positive and 140 negative from experts and, 400 positive and 400 negative truthful reviews; for the Doctor domain: 200 reviews from turkers, 32 from experts and 200 truthful opinions; for the Restaurant domain: 200 positive reviews from Turkers, 120 from experts and, 200 positive and 200 negative truthful reviews. Due to privacy policies some of the reviews originally included in the corpus are not available. Then, the version used in this work (Deception_dataset) is the one publicly available on line.[d] In order to perform the cross-domain classification we only use the positive reviews because for the Doctor domain there are not documents in the negative class. The amount of reviews for each category of Deception_dataset can be observed in the right side of Table 1.

Table 1. Corpora used for intra and cross-domain classification.

Opinion Spam			**Deception_dataset**			
Category	Turker	Truthful	Category	Turker	Expert	Truthful
Positive	400	400	*Hotel*	400	140	400
			Doctor	357	—	200
Negative	400	400	*Restaurant*	200	—	200

5. Experimental Study

In order to evaluate our proposal, we have performed the experimental study on the available opinion spam corpora. We first show the different experiments made on the single domain corpus with the different features, a study about the curse of

[d]http://web.stanford.edu/~jiweil/Code.html

dimensionality of our proposed representation and a comparison of our results with those published previously. After, we show the cross-domain experimental study and a comparison of performance with previously presented works. It is worth mentioning that for the comparisons we have employed the same measures used by the authors in their published works: in some of them they used accuracy and F-measure in others. Finally we compare graphically the amount of features used by our proposal and the state-of-the-art approaches for the classification of deceptive opinions.

5.1. *Intra-domain classification*

5.1.1. *Experiments*

We have obtained the representations of the reviews in the Opinion Spam corpus considering the features described in Section 3. For all, we have used the term frequency-inverse document frequency (tf-idf) weighting scheme. The only text preprocessing made was to convert all words to lowercase characters. Naïve Bayes and SVM algorithms in Weka[13] were used to perform the classification. We only show the results obtained with SVM because its performance was the best.[4] For all the experiments we have performed a 10 fold cross-validation procedure in order to study the effectiveness of the SVM classifier with the different representations. For simplicity, we have used LibSVM[e] which implements a C-SVC version of SVM with a radial basis function. We have run the classifier with the default parameters. The values reported in the tables correspond to the macro average F-measure as it is reported in Weka. Tables 2, 3 and 4 show the F-measure obtained with the features proposed for the Opinion Spam corpus.

Table 2 considers only the positive reviews (800 documents). In the first part of the table, we can observe the F-measure obtained with the features 3 and 4 grams in tokens and, articles, pronouns and verbs extracted from LIWC2007 (referenced as LIWC for simplicity). With the sentiment-based features (POSNEG in the table) we did not obtain good results; for that reason these are not included in the first part of the table. In the second part of the table, the combination of the different features was used as representation of the reviews. The best F-measure value is in boldface. As we can observe, the best result (F-measure = 0.89) was obtained with the combination of character 4-grams in tokens and the articles, pronouns and verbs (LIWC) referenced henceforth as 4-grams+LIWC for simplicity. With the combination of 3-grams and LIWC features the F-measure is quite similar.

Table 3 shows the results obtained considering only the negative reviews (800 documents). The best result (F-measure = 0.865) was obtained with the character 4-grams in tokens plus LIWC-based features. It is interesting to note that similar results (although sightly lower) were obtained also with the character 4-grams in tokens, character 3-grams combined with LIWC features and also with the feature 4-grams+POSNEG.

[e]https://www.csie.ntu.edu.tw/~cjlin/libsvm/

Table 2. Deceptive opinions detection with SVM for positive reviews of the Opinion Spam corpus (800 opinions).

Features	**F-measure**
3-grams in tokens	0.821
4-grams in tokens	0.871
LIWC	0.697
3 + 4-grams in tokens	0.873
3-grams + POSNEG	0.871
4-grams + POSNEG	0.873
3 + 4-grams + POSNEG	0.877
3-grams + LIWC	0.883
4-grams + LIWC	**0.89**

Table 3. Deceptive opinions detection with SVM for negative reviews of Opinion Spam corpus (800 opinions).

Features	**F-measure**
3-grams in tokens	0.826
4-grams in tokens	0.851
LIWC	0.69
3 + 4-grams in tokens	0.832
3-grams + POSNEG	0.827
4-grams + POSNEG	0.851
3 + 4-grams + POSNEG	0.827
3-grams + LIWC	0.85
4-grams + LIWC	**0.865**

Table 4. Deceptive opinions detection with SVM for positive and negative reviews of the Opinion Spam corpus (1600 opinions).

Features	**F-measure**
3-grams in tokens	0.766
4-grams in tokens	0.867
LIWC	0.676
3 + 4-grams in tokens	0.854
3-grams + POSNEG	0.858
4-grams + POSNEG	0.87
3 + 4-grams + POSNEG	0.851
3-grams + LIWC	0.866
4-grams + LIWC	**0.879**

Table 4 shows the classification results considering the whole corpus, that is, the combined case of positive plus negative reviews (1600 documents). The best F-measure (0.879) was obtained, as the same as the previous cases, with character

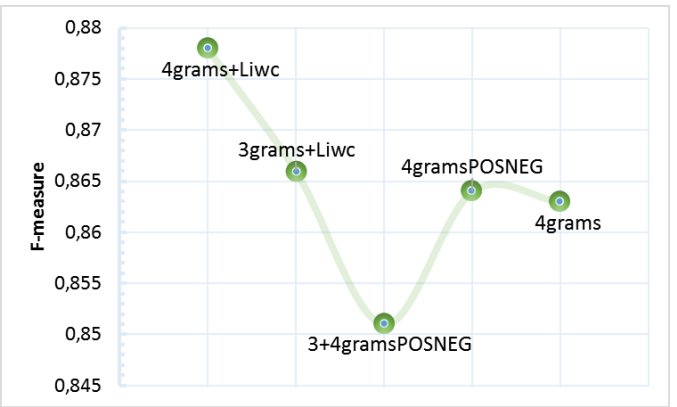

Fig. 2. Ranking of the 5 most effective features.

4-grams in tokens plus LIWC-based features. It is worth noting that the combination of character 4-grams in tokens with the POSNEG features seems to be effective when positive and negative polarities are considered together in deception detection, a fact that is not present when just one polarity is considered (see Tables 2 and 3).

As we can observe from Tables 2, 3 and 4, the differences of F-measure values are quite small. In fact, for the almost similar values like, for example, character 4-grams in tokens+LIWC compared with 3-grams+LIWC or 3+4-grams+POSNEG (see Table 2) the differences are not statistically significant. Consequently we have selected the one with highest F-measure value (character 4-grams in tokens+LIWC), but some of the other representations can be used instead. Figure 2 shows the 5 most effective features, that is, those with which we obtained the best F-measure values in classification experiments (an average over the performance obtained with the positive, negative and both together reviews). It is possible to observe that the best F-measure was obtained with character 4-grams in tokens+LIWC.

In order to analyse the set of features corresponding to character 4-grams in tokens combined with LIWC, we have calculated the Information Gain ranking. From this analysis we have observed that the set of features with highest information gain is similar for the negative polarity corpus and the corpus with the combination of both polarities reviews. The study shows that character 4-grams in tokens are in the top positions of the ranking and those reveal information related to places (*chic, chig, igan* for Chicago and Michigan cities), amenities (*floo, elev, room* for floor, elevator, room) and their characterization (*luxu, smel, tiny* for luxury, smells and tiny). From the 7th position of the ranking we can observe the first LIWC features: pronouns (*my, I, we*) and after 15th position we can observe verbs (*is, open, seemed*). Interestingly, the articles can be observed from position 68th in the ranking (*a, the*). Figure 3 illustrates the first positions of the ranking of features obtained for negative reviews.

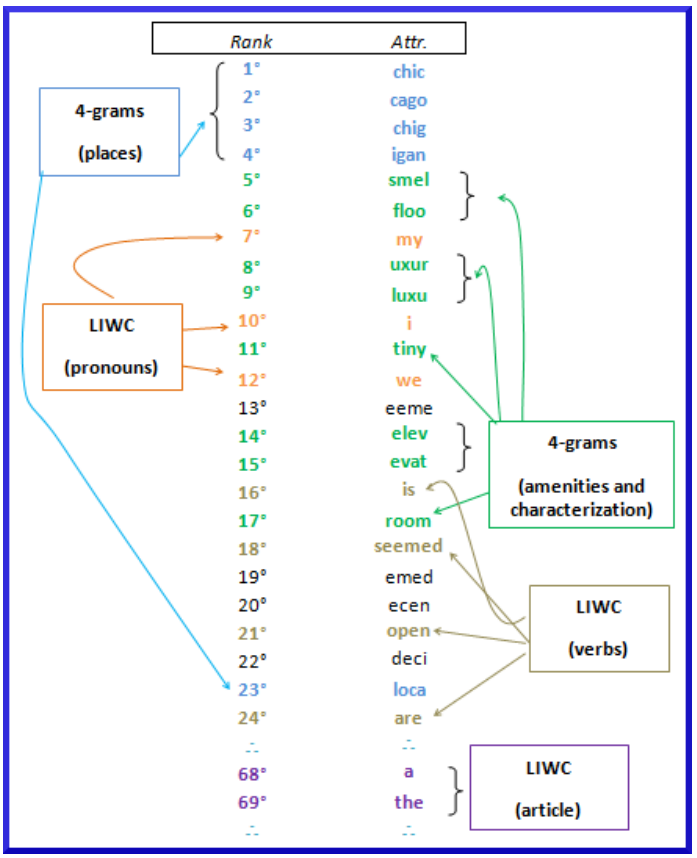

Fig. 3. Most discriminative features for negative reviews.

Considering only the positive reviews, the ranking is similar to the cases analysed before with exception of the pronouns which appear at 1st position (*my*) and at 16th position (*I, you*). This fact could indicate the presence of many opinions concerned with their own experience (good) making the personal pronouns one of the most discriminative features for positive polarity spam opinion detection. With respect to the characterization of the amenities, the adjectives observed in 4-grams in tokens have to do with positive opinions about those (*elax, amaz, good* for relax, amazing and good). Figure 4 illustrates the first positions of the ranking of features obtained for positive reviews.

Regarding the amount of features of character 4-grams in tokens combined with LIWC, we can observe that the dimensionality of this representation is lower than the standard ones (character n-grams and bag of words) but a deeper analysis of this issue should be performed in order to detect possible problems such as overfitting. Next subsection presents some discussions about that and the relation with the curse of dimensionality.

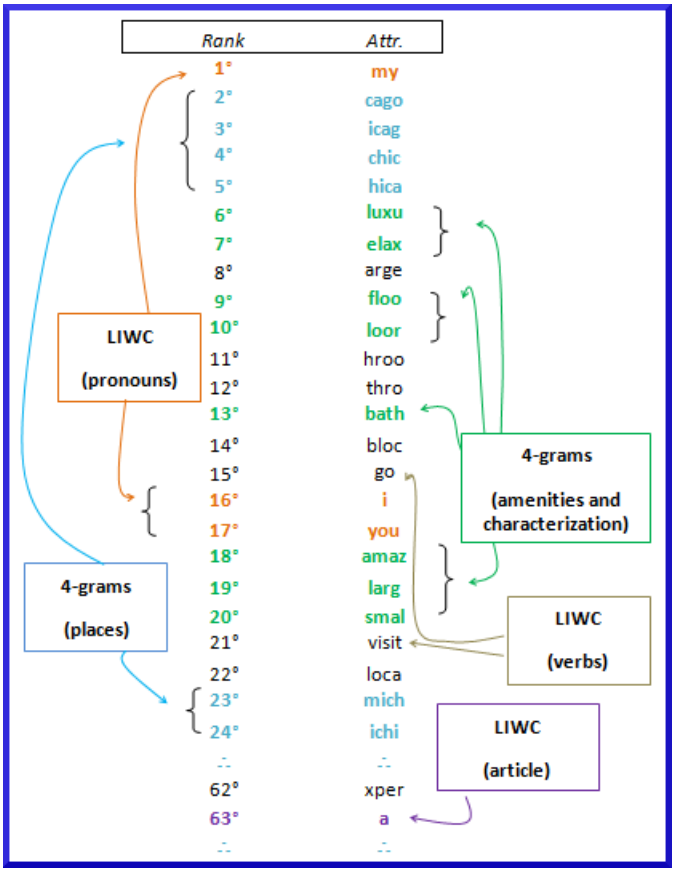

Fig. 4. Most discriminative features for positive reviews.

5.1.2. *Character 4-grams + LIWC and the curse of dimensionality*

The main goal of machine learning algorithms is to generalize and fit the data reasonably well, that is, to learn a model from examples (training set) and to perform well with new ones (test set). That generalization has important aspects to consider: how many data should be in the training set and how much knowledge we can obtain from them (the representation). The amount of data for training the classifier generally depends on their availability (the corpus) and how to use them. The representation of the data is an important issue to consider because the good performance of the classifier depends on the knowledge extracted from that. In Ref. 25, these two aspects are named global information (abundance of training set) and specific information (input dimensionality). The authors interpret the curse of dimensionality as: "too much specific information is bad and the more global information the better". Sometimes we have a classifier that fits the training data too tightly performing well on those but bad in test data (generalization problem

or overfitting). The authors in Ref. 33 relate the curse of dimensionality and the scarcity of data because sometimes the input dimensionality is high compared with the amount of training data ($d \gg n$). It is wrong to think that representations including many features are more informative and, therefore, the classifier generalizes better. Even in cases in which $d \gg n$, overfitting can be severe. In order to analyse the overfitting in the context of this work, we can obtain a decomposition of the generalization error in two values: bias and variance. Bias indicates the adequateness of the model regarding the truth. Variance indicates the sensibility of the model prediction given the training data. High variance and low bias could indicate overfitting while high bias and low variance could indicate underfitting (the classifier can not learn the underlying trend of the data). Although cross-validation experimentation can help to prevent overfitting, we study the averaged error estimation of the classification using the character 4-grams + LIWC representation showed in the previous subsection, in order to analyse if overfitting occurs in the experiments. The results showed in Tables 2 and 3 were obtained with 1533 and 1497 features respectively (default values obtained with the selected representation). Following, we perform the analysis considering more and less features than those. We use the bias-variance decomposition which consists of a N-times k fold cross-validation in order to guarantee that data will be tested N times.[40] In particular, we run the Weka implementation[39] with the default parameters (each instance is classified 10 times) using a SVM classifier (also with default parameters).

Figure 5 shows the results of error of classification, bias and variance for positive reviews, using different amount of features: 80, 400, 800, 1533, 1850 and 2500. As we can observe, the error values are low while the dimensionality is increased but with more than 1533 features these seem to be worse. Considering bias, there is a clear tendency to decrease the values which seem to avoid the underfitting phenomenon. The variance is close to zero, reaching high values when 1533 or more features are used. Combining error, bias and variance the dashed vertical line in Figure 5 shows

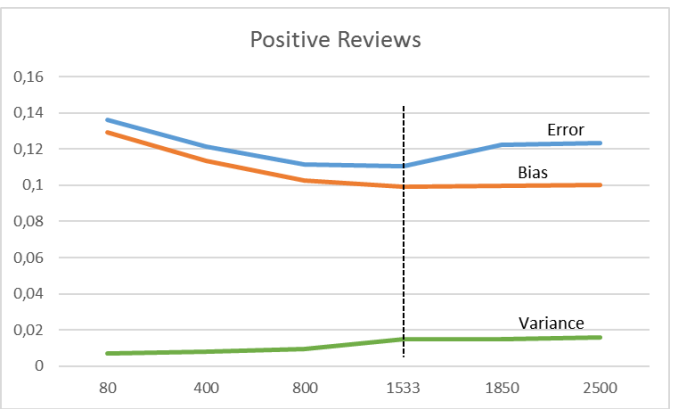

Fig. 5. Error, bias and variance for different features and positive reviews.

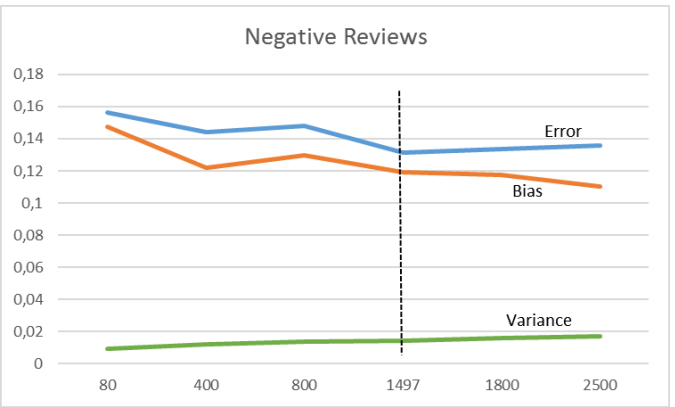

Fig. 6. Error, bias and variance for different features and negative reviews.

a trade-off between bias and variance while the error is the lowest. With that we can suppose that no overfitting (neither underfitting) is present.

Figure 6 shows a similar study considering 80, 400, 800, 1497, 1800 and 2500 features for negative reviews. The classification error reaches the lowest value when the dimensionality of the representation has 1497 attributes. Bias and variance have similar forms than in the previous case: bias tends to decrease while variance tends to increase very slowly. The trade-off is showed with a dashed vertical line in the figure. No overfitting neither underfitting is present.

As conclusion of this analysis, we can state that bias is reduced and variance is increased when the dimensionality of the representation is increased (also the complexity model). Although there is not an analytic way to find a trade-off between bias and variance, we studied the behaviour of the character 4-grams + LIWC for the classification of deceptive opinions using a bias-variance decomposition combined with the classification error. Then we are sure that the low dimensionality of this representation has not associated problems resulting adequate for the representation of reviews.

5.1.3. *Comparison of results*

For a comparison of the performance of our proposal, we compared our results with the ones of the state-of-the-art. We considered the results of five different models. Two kinds of comparisons are shown: an indirect (we could not obtain the complete set of results reported by the authors) and a direct (the authors made available the results and a statistical comparison could be performed).

In Table 5, we can observe the indirect comparison of our results with those previously presented[1,34] obtained with 10 fold cross-validation, and then, with a 5 fold cross-validation in order to make a fair comparison with the results of Ott *et al.*[31] and Feng *et al.*[10] Note that the results are expressed in terms of the accuracy

Table 5. Indirect comparison of the performance. Deceptive opinions detection for positive reviews of the Opinion Spam corpus (800 opinions).

Model	Accuracy
10 fold cross-validation	
Logistic regression[1]	70.50%
PU-learning[34]	86.69%
Our model	**89%**
5 fold cross-validation	
LIWC+grams[31]	89.8%
Profile alignment[10]	**91.3**%
Our model	89.8%

as those published by the authors; the results correspond only to positive reviews of the Opinion Spam corpus because the authors experimented with that part of the corpus only.

From Table 5, we can observe that the logistic regression approach[1] has the lowest prediction accuracy (70.50%). The accuracy of the semi-supervised model[34] is slightly lower (86.69%) than that of our model (89%), although good enough. Regarding the experiments with the 5 fold cross-validation, we obtained similar results to those of Ott *et al.*[31] and slightly lower than the ones of Feng *et al.*[10] The representation of Feng *et al.*[10] needs more than 20138 features while with our model we could obtain comparable results with a smaller representation of 1533 features (see Table 6).

In Table 6, we can observe the direct comparison with the performance on the positive and negative polarities reviews of the Opinion Spam corpus as it was obtained in Hernández Fusilier *et al.*[17] The first column shows the representation proposed, the second one shows the amount of features of the representation, the third column shows the F-measure value obtained after a 10 fold cross-validation process,

Table 6. Direct comparison of the performance for deceptive opinions detection.

Positive reviews (800 opinions)			
Model	**Features**	**F-measure**	**p-value**
Char 5-grams[17]	60797	0.90	0.094
Our model	1533	0.89	
Negative reviews (800 opinions)			
Model	**Features**	**F-measure**	**p-value**
Char 4-grams[17]	32063	0.872	0.748
Our model	1497	0.865	

and the last column shows the p-value obtained in the statistical significance test used to study the differences of performance between the character n-grams based PU-learning approach[17] and our model.

It is interesting to note that the F-measure values obtained with both approaches are quite similar for positive and negative reviews, as we can observe in Table 6. Regarding the amount of features used for each representation of the reviews, it is worth noting that our approach uses 97% and 95% less features for positive and negative reviews compared with the model of Hernández Fusilier *et al.*[17] Even using a combination of two simple set of features as character 4-grams in tokens and the LIWC-based features, the amount of attributes we used is considerably lower than the traditional character n-grams without diminishing the quality of the classification. The reason of the lower dimensionality of our representation has to do with the way in which the n-grams are obtained. The high descriptive power of character n-grams in tokens plus the information added with the LIWC-based features allow to detect deceptive opinions obtaining a good performance.

In order to determine if the differences of performance of Hernández Fusilier *et al.*[17] and our model are statistically significant, we have calculated the Mann-Whitney U-test.[26] This non-parametric test compares two unpaired groups of values without making the assumption of the normality of the samples. However, the requirements of independence of the samples, the data is continuous, ordinal and there are no ties between the groups, and the assumption that the distribution of both groups is similar in shape, are satisfied. The null hypothesis states that the samples come from the same population, that is, the classifiers performs equally well with the proposed models. We have calculated the Mann-Whitney U-test considering a 2-tailed hypothesis and significance level of 0.05. In Table 5 we can observe that the p-value obtained in the comparison of the performance of positive reviews corpus is $0.094 > 0.05$ which stands for the difference of results is not statistically significant (the p-value is not ≤ 0.05, then the null hypothesis is not rejected). The same conclusion can be obtained with respect to the results corresponding to the negative reviews corpus, for which the test obtained a p-value of $0.748 > 0.05$. From the last test we may conclude that both approaches performs similarly well. A statistical analysis of variance over the F-measure values obtained in the evaluation of Hernández Fusilier *et al.*[17] and our approach complements our performance study. This analysis can be obtained from the boxplots[f] with the distribution of F-measure values of each proposal with both polarities reviews corpora. Figures 7 and 8 illustrate this analysis. In both figures we can observe that our approach shows a higher dispersion of values, as well as the best F-measure values (0.94 for positive reviews corpus and 0.915 for negative reviews) and the minimum F-measure values (0.84 and 0.81 for positive and negative polarities, respectively) compared to the values

[f]Boxplots[36] are descriptive statistical tools for displaying information (dispersion, quartiles, median, etc.) among populations of numerical data, without any assumptions about the underlying statistical distribution of the data.

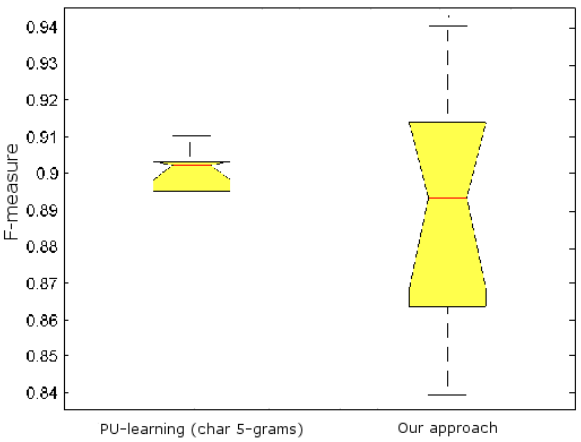

Fig. 7. Boxplot for positive reviews corpus in the direct comparison of performance.

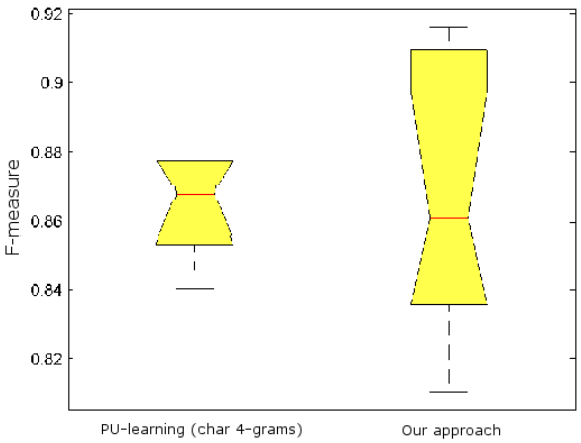

Fig. 8. Boxplot for negative reviews corpus in the direct comparison of performance.

obtained in Hernández Fusilier *et al.*[17] However, the median values obtained with both models are quite similar, reason why we conclude that there is not statistical difference of performance as it was showed with the statistical significance test.

5.2. *Cross-domain classification*

5.2.1. *Experiments*

In order to study the impact of the character 4-grams+LIWC features on cross-domain classification, we carried out some experiments using the Deception_dataset. The underlying idea involves to obtain a model trained with deceptive opinions of a particular domain and test it with data with a different topic (possibly a different distribution).

Table 7. Metrics for cross-domain classification of the Deception_dataset.

Train/Test	Baseline	Accuracy	Averaged F-measure	Deceptive class Precision	Recall	F-measure
Hotel/Restaurant	56%	65.25%	0.64	0.62	0.79	0.69
Hotel/Doctor	58%	63.97%	0.5	0.64	1	0.78
Doctor/Restaurant	56%	57.46%	0.57	0.56	0.69	0.62
Doctor/Hotel	54%	57.44%	0.42	0.57	1	0.73
Restaurant/Doctor	58%	63.97%	0.5	0.64	1	0.78
Restaurant/Hotel	54%	65.85%	0.66	0.73	0.64	0.68

Table 7 shows the classification results obtained with a similar configuration than the one used for the intra-domain experiment: character 4-grams in tokens + LIWC features with tf-idf weighting schema for the representation of opinions, and the LibSVM classifier with the default parameters of the Weka tool. The first column indicates the domains used for training/testing the classifier, the second one indicates the baseline, the third column indicates the accuracy of the classification, the fourth column indicates the averaged F-measure (over truth and deceptive classes) and the next three columns indicate precision, recall and F-measure of the deceptive class. The baseline was calculated through a simulation implementing a Monte Carlo method. After 10000 independent runs with each domain, we selected the values for which the 99% of the simulations did not exceed those percentages of correct answers (opinions correctly classified) and used those as baseline. The amounts of features obtained with our proposal for the experiments were 1008, 1001 and 1042 attributes when Hotel, Doctor and Restaurant were used respectively for the training.

As we can observe from Table 7 the accuracy values obtained in each experiment are around the 60% and in all cases these exceed the baseline. In particular, the highest values (and quite similar) were obtained when the classifier was trained with hotel reviews and tested with those of restaurants, and inversely when the classifier was trained with restaurants and tested with reviews of hotels (more than 65% of accuracy for both cases). This could be because opinions about hotels and restaurants have many words in common due to these domains share some properties as the characterization of the places. The lowest values of accuracy were obtained using the reviews of doctors for training and testing the classifier with hotels and restaurants reviews (around 57%). This could indicate that the learned model with words related to the doctor domain does not help much to discriminate truthful from deceptive reviews of hotels and restaurants. The averaged F-measure values are not good with the exception of the case Hotel/Restaurant and vice versa. The worst was for the case Doctor/Hotel (0.42), following the cases in which the Doctor domain was used for testing. The bad values of the averaged F-measure maybe could indicate that the performance of the classifier for the truth class is quite low or maybe similar than the observed for the deceptive class. Regarding

F-measure for the deceptive class, we can observe that values are good although this is because recall values are high. For the cases is which Doctor is the test domain, F-measure is the highest. Inversely, when Doctor is used for training, F-measure is lower (around 0.7) with the lowest values of precision. Interestedly, there are three cases for which recall values are 1 which indicates that the classifier retrieved all relevant results.

In order to analyse the more significant attributes for each domain, we have calculated the information gain ranking of each one. In general, in the first positions of the rankings we can observe character 4-grams in tokens revealing information about amenities (*ocat, bath, floo* for locate, bathroom and floor) in the Hotel domain and, location and characteristic of personality (*offi, frie* for office and friend) in the Doctor and Restaurant domains. Then, pronouns, articles and verbs together with character 4-grams in tokens complete the rankings.

5.2.2. *Comparison of results*

In order to study the complexity of the cross-domain classification with the Deception_dataset, we first show in Table 8 the accuracy values obtained with a SVM classifier using character 4-grams+LIWC features in intra-domain classification versus cross-domain classification. The left side of the table shows the comparison of results with those of Li *et al.*[22] for intra-domain classification (truthful vs turker reviews with 10 fold cross-validation procedure). The right side of the table illustrates the best results obtained for our proposal for cross-domain scenario (see Table 7 for a complete description). We can not compare these results with the published in Ref. 22 because the authors used a different version of the corpus. The first 3 columns of Table 8 show the domain, amount of features used and the baseline determined as we explained in the previous subsection. The fourth and fifth columns show the accuracy values obtained with character 4-grams+LIWC features and the unigrams of Li *et al.* for the comparison of single domain experiments. The last two columns show the best results obtained in cross-domain classification (from Table 7).

From Table 8, we can observe that both proposals outperformed the baseline and we obtained good results for the intra-domain experiment for the Hotel and the Doctor domains. For the Restaurant domain, the unigrams used in Ref. 22 obtained slightly better results than ours. Although, as was expected, the performance

Table 8. Accuracy values for intra and cross-domain classification.

Intra-domain Classification					**Cross-domain Classification**	
Domain	Features	Baseline	Our model	Unigrams[22]	Test-domain	Our model
Hotel	1277	54%	**88.25%**	81.8%	*Restaurant*	65.25%
Doctor	1341	58%	**82.07%**	74.5%	*Restaurant*	57.46%
Restaurant	1263	56%	78.85%	**81.7%**	*Hotel*	65.85%

Table 9. Hernández Fusilier's version of the Deception_dataset used for cross-domain classification.

	Deception_dataset	
	Turker	Truthful
Hotel	400	140
Doctor	90	200
Restaurant	50	200

obtained is lower for the cross-domain scenario, the character 4-grams+LIWC features allowed to obtain reasonable results.

As we stated in Section 4 we could not compare the cross-domain experiments directly with those presented in Li *et al.*[22] because we did not have access to the complete corpus but only to its publicly available version. Therefore, we select other results to compare the performance of our proposal.

Hernández Fusilier[15] used a similar version of the Deception_dataset in a cross-domain experiment. The author considered just a subset of the Deception_dataset corpus (shown in Table 9) and reported the performance of the Naïve Bayes classifier using the PU-learning variant proposed in Hernández Fusilier *et al.*[17] with char 4-grams. Besides, the author only shows the results of his proposal training the model with the Hotel domain and testing with Restaurant and Doctor domains. In order to make a fair comparison, we performed the same experiments reported by the author and with the same subset used in Hernández Fusilier.[15]

The results obtained are shown in Table 10. The first column shows the domains used for training/testing the classifiers, the second and third columns show the model evaluated and the baseline; the averaged (over both classes) F-measure value is in the fourth column and the metrics only for deceptive class are shown in the last three columns. In order to use a clear baseline, a Monte Carlo simulation was performed with 10000 independent executions. The results for each domain showed that less than 1% of simulations exceeded the 0.57 of F-measure, then that value was used as a baseline. Regarding the amount of features corresponding to each representation, our approach uses 96% less attributes than that of Hernández Fusilier,[15] that is, character 4-grams+LIWC has 1020 attributes while char 4-grams[15] has

Table 10. Cross-domain classification comparison using the subset of the Deception_dataset.

			Averaged	**Deceptive class**		
Train/Test	**Model**	**Baseline**	**F-measure**	**Precision**	**Recall**	**F-measure**
Hotel/Restaurant	Char 4-grams[15]	0.57	**0.83**	**0.54**	0.72	**0.62**
Hotel/Restaurant	Our model	0.57	0.75	0.36	**0.9**	0.52
Hotel/Doctor	Char 4-grams[15]	0.57	0.47	0.33	0.7	0.45
Hotel/Doctor	Our model	0.57	**0.58**	**0.35**	**0.76**	**0.48**

26210. All values of averaged F-measure obtained with our proposal outperform the baseline, not observing the same for the case Hotel/Doctor with Hernández Fusilier proposal which is quite lower. The averaged F-measure value obtained for training the model with the Hotel domain and testing with reviews of restaurants is slightly better with char 4-grams than character 4-grams+LIWC features (although our model needs only 1k features versus the 26k of the PU-learning one). The low dimensionality of our representation could affect the classification task when cross-domain experiments are considered. In particular if the training and testing domains are quite similar, the fact of the low dimensionality of the set of features could remove discriminative attributes needed in this kind of classification. However, when domains are not similar as Hotel and Doctor, the character 4-grams+LIWC features seem to capture important information to obtain an adequate model. The averaged F-measure value obtained with our approach is 20% better than that obtained in Hernández Fusilier[15] with char 4-grams for the Hotel/Doctor experiment. If metrics for deceptive class are observed, similar conclusions can be obtained. For the case Hotel/Restaurant the precision of our proposal is quite lower than that of Hernández Fusilier, the same that the corresponding F-measure. For the case Hotel/Doctor we can observe the opposite, that is, the metrics are higher when our proposal is used. It is interesting to note that the recall obtained with character 4-grams+LIWC in all cases is higher than those of Hernández Fusilier which means that our approach classified correctly most relevant reviews.

5.3. *Summary*

Figure 9 shows the amount of features used by our proposal compared with those of Hernández Fusilier[15,17] in order to illustrate the suitability of our low dimensionality representation for the classification of deceptive opinions. It is possible to observe that for intra domain classification, our representation uses around 1.5K of features compared with 60K and 30K for representing positive and negative reviews respectively, in Hernández Fusilier *et al.*[17] Regarding the cross-domain classification, the difference on the dimensionality is lower than the previous cases but

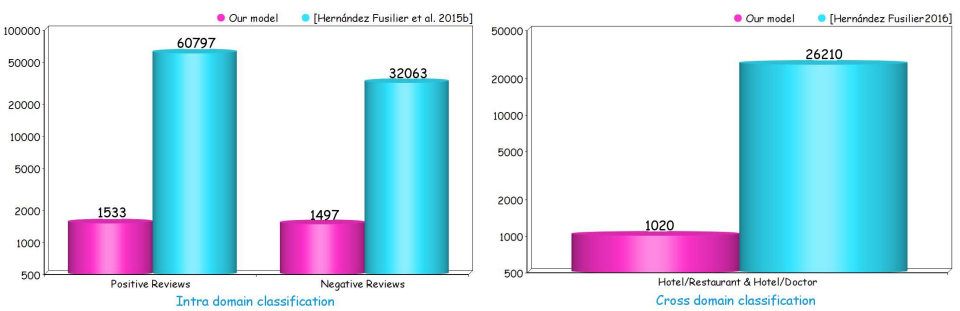

Fig. 9. Amount of features used for opinion spam classification.

considerable: 1K versus 26K of Hernández Fusilier.[15] The performance of the classifier (intra and cross-domain scenarios) with character 4-grams in tokens+LIWC was better in some cases and comparable in others, regarding the performance of published methods. It is worth mentioning that our model obtained high recall values in all experiments which shows that the relevant reviews were retrieved. Then, we conclude that character 4-grams in tokens+LIWC is an interesting representation due the low dimensionality and the good performance obtained for spam detection in intra and cross-domain classifications.

6. Conclusions and Future Work

In this work we have investigated how different features contribute to model deception clues. Character n-grams in tokens showed to capture the content and the writing style of the reviews allowing to differentiate truthful from deceptive opinions. On the contrary, sentiment-based features did not help us to discriminate deceptive opinions. We have also employed as features information extracted from LIWC as pronouns, articles and verbs. These features combined with character 4-grams in tokens were finally employed for a low dimensionality representation of the reviews. For the intra-domain experimental study we compared the results obtained using SVM with character 4-grams in tokens with LIWC-based features, with state-of-the-art approaches. Our results were better in most of the cases, although no statistically significant difference was found. What is important to highlight is that our model allows to work with a lower dimensionality representation that makes it more efficient. We also performed cross-domain experiments with the aim of validate our model in a realistic scenario where deceptive opinions may be available in a domain but not in another one. The low dimensionality of character 4-grams in tokens together with LIWC-based features allowed to obtain comparable results to state-of-the-art ones but employing only 1K features instead of 26K. As future work we plan to investigate emotion features[5,6] in the task of deceptive opinion detection. Moreover, we are interested in testing our model with other corpora related to opinion spam as the one recently proposed in Fornaciari and Poesio.[12] We also plan to perform a deeper study about the importance of each feature on deceptive detection. In particular, we think that hybrid feature selection methods like the combination of wrapper methods and some scoring measures[27] would be useful.

Acknowledgements

This publication was made possible by NPRP grant #9-175-1-033 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

1. S. Banerjee and A. Y. K. Chua, Dissecting genuine and deceptive kudos: the case of online hotel reviews, *Int. J. Advanced Computer Science and Applications* (*IJACSA*), Special Issue on Extended Papers from Science and Information Conference (2014), pp. 28–35.
2. S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit* (O'Reilly, 2009).
3. J. K. Burgoon, J. P. Blair, T. Qin, and J. F. Nunamaker Jr., Detecting deception through linguistic analysis, in H. Chen, R. Miranda, D. D. Zeng, C. Demchak, J. Schroeder, and T. Madhusudan (eds.), *Intelligence and Security Informatics*, Lecture Notes in Computer Science, Vol. 2665 (Springer Berlin Heidelberg, 2003), pp. 91–101.
4. L. Cagnina and P. Rosso, Classification of deceptive opinions using a low dimensionality representation, in *Proc. 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (Association for Computational Linguistics, 2015), pp. 58–66.
5. E. Cambria, Affective computing and sentiment analysis, *IEEE Intelligent Systems* **31**(2) (2016) 102–107.
6. E. Cambria and A. Hussain, Sentic computing, *Cognitive Computation* **7**(2) (2015) 183–185.
7. W. B. Cavnar and J. M. Trenkle, N-gram-based text categorization, *Proc. SDAIR-94, 3rd Annual Symp. Document Analysis and Information Retrieval*, Ann Arbor, MI, **48113**(2) (1994), pp. 161–175.
8. S. Feng, R. Banerjee, and Y. Choi, Syntactic stylometry for deception detection, in *Proc. 50th Annual Meeting of the Association for Computational Linguistics* (The Association for Computer Linguistics, 2012), pp. 171–175.
9. S. Feng, L. Xing, A. Gogar, and Y. Choi, Distributional footprints of deceptive product reviews, in *Proc. Sixth Int. AAAI Conf. Weblogs and Social Media* (The AAAI Press, 2012), pp. 98–105.
10. V. W. Feng and G. Hirst, Detecting deceptive opinions with profile compatibility, in *Proc. 6th Int. Joint Conf. Natural Language Processing* (The Association for Computer Linguistics, 2013), pp. 338–346.
11. E. Fitzpatrick, J. Bachenko, and T. Fornaciari, *Automatic Detection of Verbal Deception*, Synthesis Lectures on Human Language Technologies (Morgan & Claypool Publishers, 2015).
12. T. Fornaciari and M. Poesio, Identifying fake amazon reviews as learning from crowds, in *Proc. 14th Conf. European Chapter of the Association for Computational Linguistics* (Association for Computational Linguistics, 2014), pp. 279–287.
13. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, The weka data mining software: an update, *ACM SIGKDD Explorations Newsletter* **11**(1) (2009) 10–18.
14. J. T. Hancock, L. E. Curry, S. Goorha, and M. Woodworth, On lying and being lied to: a linguistic analysis of deception in computer-mediated communication, *Discourse Processes* **45**(1) (2008) 1–23.
15. D. H. Fusilier, *Detección de opinion spam usando PU-Learning*, Phd thesis, Universitat Politècnica de València (2016).
16. D. H. Fusilier, M. Montes-y-Gómez, P. Rosso, and R. G. Cabrera, Detecting positive and negative deceptive opinions using PU-learning, *Information Processing & Management* **51**(4) (2015) 433–443.

17. D. H. Fusilier, M. Montes-y-Gómez, P. Rosso, and R. G. Cabrera, Detection of opinion spam with character n-grams, in A. Gelbukh (ed.), *Computational Linguistics and Intelligent Text Processing*, Lecture Notes in Computer Science, Vol. 9042 (Springer International Publishing, 2015), pp. 285–294.
18. N. Jindal and B. Liu, Opinion spam and analysis, in *Proc. 2008 Int. Conf. Web Search and Data Mining* (*WSDM '08*) (ACM, 2008), pp. 219–230.
19. N. Jindal, B. Liu, and E. Lim, Finding unusual review patterns using unexpected rules, in J. Huang, N. Koudas, G. J. F. Jones, X. Wu, K. Collins-Thompson, and A. An (eds.), *Proc. 19th ACM Int. Conf. Information and knowledge management*, (ACM, 2010), pp. 1549–1552.
20. V. Keselj, F. Peng, N. Cercone, and C. Thomas, N-gram-based author profiles for authorship attribution, In *Proceedings of the Conference Pacific Association for Computational Linguistics, PACLING'03* (Dalhousie University, 2003), pp. 255–264.
21. J. Li, C. Cardie, and S. Li, TopicSpam: a topic-model based approach for spam detection, in *Proc. 51 Annual Meeting of the Association for Computational Linguistics* (The Association for Computer Linguistics, 2013), pp. 217–221.
22. J. Li, M. Ott, C. Cardie, and E. H. Hovy, Towards a general rule for identifying deceptive opinion spam, in *Proc. Conf. of the Association for Computational Linguistics* (The Association for Computer Linguistics, 2014), pp. 1566–1576.
23. Y. Lin, T. Zhu, H. Wu, J. Zhang, X. Wang, and A. Zhou, Towards online anti-opinion spam: spotting fake reviews from the review sequence, in *Proc. 2014 IEEE-ACM Int. Conf. Advances in Social Networks Analysis and Mining* (*ASONAM*) (IEEE, 2014), pp. 261–264.
24. B. Liu, W. S. Lee, P. S. Yu, and X. Li, Partially supervised classification of text documents, in *Proc. Nineteenth Int. Conf. Machine Learning* (Morgan Kaufmann Publishers Inc., 2002), pp. 387–394.
25. R. Liu and D. F. Gillies, Overfitting in linear feature extraction for classification of high-dimensional image data, *Pattern Recognition* **53**(C) (2016) 73–86.
26. H. B. Mann and D. R. Whitney, On a test of whether one of two random variables is stochastically larger than the other, *The Annals of Mathematical Statistics* **18**(1) (1947) 50–60.
27. E. Montañés, J. R. Quevedo, E. F. Combarro, I. Díaz, and J. Ranilla, A hybrid feature selection method for text categorization, *Int. J. Uncertainty, Fuzziness and Knowledge-Based Systems* **15**(2) (2007) 133–151.
28. A. Mukherjee, B. Liu, J. Wang, N. Glance, and N. Jindal, Detecting group review spam, in *Proc. 20th Int. Conf. Companion on World Wide Web* (ACM, 2011), pp. 93–94.
29. M. L. Newman, J. W. Pennebaker, D. S. Berry, and J. M. Richards, Lying words: predicting deception from linguistic styles, *Personality and Social Psychology Bulletin* **29**(5) (2003) 665–675.
30. M. Ott, C. Cardie, and J. T. Hancock, Negative deceptive opinion spam, in *Proc. 2013 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (The Association for Computational Linguistics, 2013), pp. 497–501.
31. M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, Finding deceptive opinion spam by any stretch of the imagination, in *Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (ACM, 2011), pp. 309–319.
32. J. W. Pennebaker, C. K. Chung, M. Ireland, A. Gonzales, and R. J. Booth, The development and psychometric properties of LIWC2007, in *LIWC webpage* (LIWC.net, 2007), pp. 1–22, http://www.liwc.net/LIWC2007LanguageManual.pdf.

33. S. J. Raudys and A. K. Jain, Small sample size effects in statistical pattern recognition: Recommendations for practitioners, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13**(3) (1991) 252–264.
34. Y. Ren, D. Ji, and H. Zhang, Positive unlabeled learning for deceptive reviews detection, in A. Moschitti, B. Pang, and W. Daelemans (eds.), *Proc. 2014 Conf. Empirical Methods in Natural Language Processing* (*EMNLP 2014*) (The Association for Computational Linguistics, 2014), pp. 488–498.
35. E. Stamatatos, On the robustness of authorship attribution based on character n-gram features, *Journal of Law and Policy* **21**(2) (2013) 421–439.
36. J. W. Tukey, *Exploratory Data Analysis* (Pearson Education, Inc., Massachussets, USA, 1977).
37. A. Šilić, J. Chauchat, B. Dalbelo Bašić, and A. Morin, N-grams and morphological normalization in text classification: a comparison on a croatian-english parallel corpus, in J. Neves, M. F. Santos, and J. M. Machado (eds.), *Progress in Artificial Intelligence*, Lecture Notes in Computer Science, Vol. 4874 (Springer, Berlin, Heidelberg, 2007), pp. 671–682.
38. G. Wang, S. Xie, B. Liu, and P. S. Yu, Identify online store review spammers via social review graph, *ACM Transactions on Intelligent Systems and Technology* **3**(4) (2012) 61.
39. G. I. Webb. Multiboosting: A technique for combining boosting and wagging, *Machine Learning* **40**(2) (2000) 159–196.
40. G. I. Webb and P. Conilione, Estimating bias and variance from data (2002).
41. Z. Wei, D. Miao, J. Chauchat, and C. Zhong. Feature selection on Chinese text classification using character n-grams, in G. Wang, T. Li, J. W. Grzymala-Busse, D. Miao, and Y. Y. Yao (eds.), *3rd Int. Conf. Rough Sets and Knowledge Technology* (*RSKT 08*), Lecture Notes in Computer Science, Vol. 5009 (Springer, Berlin, Heidelberg, 2008), pp. 500–507.