Soft Comput (2017) 21:585–595
DOI 10.1007/s00500-016-2409-2

FOCUS

Cross-domain deception detection using support vector networks

Ángel Hernández-Castañeda[1] · Hiram Calvo[1] · Alexander Gelbukh[1] · Jorge J. García Flores[2]

Published online: 2 November 2016

Abstract Our motivation is to assess the effectiveness of support vector networks (SVN) on the task of detecting deception in texts, as well as to investigate to which degree it is possible to build a domain-independent detector of deception in text using SVN. We experimented with different feature sets for training the SVN: a continuous semantic space model source represented by the latent Dirichlet allocation topics, a word-space model, and dictionary-based features. In this way, a comparison of performance between semantic information and behavioral information is made. We tested several combinations of these features on different datasets designed to identify deception. The datasets used include the DeRev dataset (a corpus of deceptive and truthful opinions about books obtained from Amazon), OpSpam (a corpus of fake and truthful opinions about hotels), and three corpora on controversial topics (abortion, death penalty, and a best friend) on which the subjects were asked to write an idea contrary to what they really believed. We experimented with one-domain setting by training and testing our models separately on each dataset (with fivefold cross-validation), with mixed-domain setting by merging all datasets into one large corpus (again, with fivefold cross-validation), and with cross-domain setting: using one dataset for testing and a concatenation of all other datasets for training. We obtained an average accuracy of 86% in one-domain setting, 75% in mixed-domain setting, and 52 to 64% in cross-domain setting.

Keywords Deception detection · Continuous semantic space model · Word-space model · Linguistic inquiry and word count · Support vector networks

1 Introduction

Nowadays, users undertake a variety of online activities such as purchasing and selling items, disseminating ideas via blogs, and exchanging information in general. Such information is not always reliable: some people use Internet to transmit information for the purpose of manipulating and deceiving other users. For instance, when a user wants to buy an item online, the main way for them to know whether the product is good or not is to read the section of opinions about it on the seller's Web page. Such opinions have been shown to have a big impact on the final decision of acquiring or not the item. Because of this, some sellers hire people to write positive opinions in order to increase the sales of a product, even if those people do not have a real idea about the quality of the item. In other cases, deceptive opinions aim to discredit products offered by the competitors.

Apart from deceptive texts written to manipulate buying decisions of the users, there are also deceptive texts that intend to change the opinion or the viewpoint of people about

Communicated by H. Ponce.

✉ Hiram Calvo
hcalvo@cic.ipn.mx

Ángel Hernández-Castañeda
ahernandez_a12@sagitario.cic.ipn.mx

Alexander Gelbukh
gelbukh@gelbukh.com

Jorge J. García Flores
jgflores@lipn.univ-paris13.fr

[1] Instituto Politécnico Nacional, Center for Computing Research CIC-IPN, Av. J.D. Bátiz e/ M.O. de Mendizábal, 07738 Mexico City, Mexico

[2] Laboratoire d'Informatique de Paris Nord, CNRS (UMR 7030), Université Paris 13, Sorbonne Paris Cité, 93430 Villetaneuse, France

a certain subject, such as a political candidate or an issue of public debate.

This makes the study of detecting deceptive texts very important. The task can be defined as identifying those written opinions in which the author aims to transmit information that he or she does not believe in (Keila and Skillicorn 2005). Studies on deceptive texts have empirically proven that truthful communication is qualitatively different from deceptive communication (Ekman 1989; Twitchell et al. 2004). Because of this, a number of projects have been launched with the aim of identifying deceptive texts as accurately as possible. For this, various datasets have been created. Such datasets consist of texts labeled as truthful or deceptive. In a machine-learning approach, a part of the texts in the dataset is used as the training set for a classifier and the remainder as a test set. Thus, a direct comparison between different classifiers and feature selection methods is possible by applying them to the same dataset.

In this work, we address two research questions. First, we assess the appropriateness of support vector networks (SVN) for the task of classification of deceptive texts as precisely as possible. Second, given that features based on LDA showed good performance when they were evaluated on each dataset separately (see Sect. 4.1), we explore whether a feature set can be sufficiently general to be used for classifying a dataset on a different topic from the topic of the dataset used for training, which would allow creating domain-independent general-purpose deception text detectors.

For this, we generated features by using various methods, such as the latent Dirichlet allocation (LDA), the linguistic inquiry and word count (LIWC) method, and a word-space model (WSM), as well as combinations of features generated with such methods. To prove the efficiency of each method, we use three datasets on different topics, specifically: OpSpam, which consists of opinions about hotels, DeRev, which consists of opinions about books bought on Amazon, and the Controversial Topics dataset, which is composed of opinions on three topics (abortion, death penalty, and best friend). Based on datasets collected, we investigate which method is better in one-domain setting, where both the training and test sets are on the same topic, in a mixed-domain setting, where both training and test sets are on a mixture of topics, and in a cross-domain setting, where the training and test sets are on different topics.

With these experiments, we evaluate the possibility of using existing datasets to detect deceptive texts on a topic for which there is no dataset available, that is, the possibility of developing a general-purpose, domain-independent deceptive text detector.

The paper is organized as follows. In Sect. 2, we discuss state-of-the-art approaches to detection of deceptive texts. In Sect. 3, we give a detailed description of feature sources (Sect. 3.1) and deception detection datasets (Sect. 3.2) we used. In Sect. 4, we describe our experimental setup and discuss our results. Finally, we draw our conclusions in Sect. 5.

2 Related work

To test performance of models for detecting deception in text, several labeled datasets have been developed. As a result, datasets were created in different ways. Gokhman et al. (2012) introduced two general ways for developing datasets: sanctioned and unsanctioned deception. In first participants are asked to lie and in second participants lie on their own. These datasets allow making a comparison among the models' performance.

To create models that lead to detect deception, different sources of features are used. These sources sometimes are combined to achieve a more accurate classification. Bag-of-words (BoW) is a common approach used to generate features for representing documents; this approach disregards grammar and even word order but keeps counting the number of instances of each word. BoW approach includes single words and n-grams. On the other hand, features based on writing style are also used. Unlike BoW approach, linguistic style considers the context to the words. Additionally, certain general deception cues are sought for detecting deception (DePaulo et al. 2003), for example, the use of unique words, self-references, modifiers, among others.

Techniques similar to BoW are based on different kinds of elements extracted from text, such as words, syllables, phonemes, letters, etc. In this way, Hernández Fusilier et al. (2015) compared word n-grams with letter n-grams. The latter shown to yield a better performance on the classified dataset. However, even though n-grams alone give acceptable results, usually they are mixed with other NLP techniques; such combined feature sets often improve the results.

Another method for representing documents is to use handcrafted dictionaries. Newman et al. (2003), for example, by analyzing LIWC's word categories, found that liars use fewer self-references and use more negative emotion words. This work laid the foundation for the LIWC tool to be widely used by other researchers (Schelleman-Offermans and Merckelbach 2010; Toma and Hancock 2012).

For instance, Mihalcea and Strapparava (2009) used the LIWC tool to discover dominant classes of deceptive texts. The authors classified a corpus of deceptive and truthful texts on controversial topics such as abortion, death penalty, and a best friend. In a similar study, Pérez-Rosas and Mihalcea (2014a) attempted to classify texts on the same topics but in different languages, such as Spanish texts written by native speakers from Mexico, English texts written by speakers from the USA, and English texts written by speakers from India.

Following the work by Mihalcea and Strapparava, Almela et al. (2012) conducted a study to detect deceptive texts written in Spanish. The authors collected a new dataset with topics on homosexual adoption, bullfighting, and feelings about a best friend. One hundred deceptive documents and one hundred truthful ones were collected for each topic, of 80 words per document on average. Distinct LIWC dimensions were used to achieve a more accurate classification by using a support vector machine (SVM).

Deception detection has been applied in different particular aspects. Williams et al. (2014) compared lies told by children and lies told by adults. The authors aimed to detect deception in courts where children testified. To generate the dataset, 48 children and 28 adults were chosen; half of the children and adults told lies and half of them told truth. Thus, the authors used the LIWC tool for generating samples for classification. Research findings showed existence of significant differences between truthful and deceptive texts, which mainly involve linguistic variables such as singular self-references (e.g., I, my, me), plural self-references (e.g., we, our, us), and positive and negative emotions. In addition, results showed that such linguistic variables were found in distinct proportion depending on whether the lie was told by a child or an adult.

In (Hauch et al. 2012) several works of deceptive text identification were analyzed. Most of them are based on documents processed by computer programs; more specifically, documents were mainly represented based on the LIWC tool. Research findings showed that liars use certain linguistic categories at a different rate than the truth-tellers.

BoW and dictionary methods have shown good performance; however, in the effort to improve results, the context of words has been also taken into account, for example, by analyzing the syntactic relations between the words using dependency trees (Feng et al. 2012; Xu and Zhao 2012). In general, the use of syntactic relations has not shown an outstanding performance in the task of classifying deceptive text. However, combining this method with a BoW approach can improve results.

In some cases, not only information of words or syntactic structures in the text is available, but also additional information supplied by the source from which the texts were extracted. Fornaciari and Poesio (2014) collected fake and real opinions from the Amazon Web site. The authors took into account, for example, information on who bought the book in question and who did not: people who bought the book and wrote their opinion had higher credibility than people who did not buy the book. This kind of extra information is rarely available, so we did not focus on this kind of features in the present study.

Pérez-Rosas and Mihalcea (2014b) collected features using different approaches, such as part-of-speech (PoS) tags, context-free grammars (CFG), unigrams, and LIWC, as well as combinations of these features. The authors achieved accuracy between 60 and 70% predicting whether a person of feminine or masculine gender had written a deceptive text. Research findings showed that the use of PoS tags and CFG did not significantly improve accuracy as compared with unigrams and LIWC. This suggests that BoW- and dictionary-based approaches give performance similar to that of linguistic style approaches.

Combining different methods is a common method in generating accurate models used in deception detection. This is done to exploit the advantages of each approach and thus improve the accuracy of classification. However, to our knowledge getting a universal domain deception detector has been scantily studied.

3 Deception detection

In this section we present our method for deception detection. First, in Sect. 3.1 we present our model for deception detection using support vector networks. Next, in Sect. 3.2 we describe the different datasets we will use for evaluation. Finally, in Sect. 3.3 we detail the different feature sources we will use.

3.1 Support vector networks

Support vector networks have been widely applied to solve tasks in different areas (Petković et al. 2014a, b; Shamshirband et al. 2014; Altameem et al. 2015; Gani et al. 2016; Kisi et al. 2015; Mohammadi et al. 2015a, b; Olatomiwa et al. 2015a, b; Piri et al. 2015; Protić et al. 2015; Shamshirband et al. 2015a, b; Al-Shammari et al. 2016a, b; Gocic et al. 2016; Jović et al. 2016a, b; Shamshirband et al. 2016a, b; Shenify et al. 2016).

For generating of a model that more closely represent the deception, we propose using support vector networks with an attribute selection (Guyon and Elisseeff 2003). The latter is important for removing repetitive and irrelevant features.

In the process of feature selection, we used the WEKA (Hall et al. 2009) tool with an evaluator based on correlation proposed by Hall (1999). Furthermore, we chose a search criterion based on hill climbing with backtracking. Such combination showed a significant increase in accuracy. We found that binarizing the feature vectors gave a more accurate model for deceptive texts detection in the present experimental setup. The same process of attribute selection, explained above, was conducted in all our experiments.

The procedure for feature vectors generation is as follows:

- To form vectors of features by using a word-space model (WSM), first, we obtained lemmas and kept stop words. Secondly, a list of all words without repeating (types) found in the texts set (as deceptive texts as truthful texts) was generated. Next, given a document and a list of types, if the current word of the type list is contained in the document, then the feature value was converted into one, in other case was converted into zero.

- LDA shows as result vectors of features with real-type values (probabilities of belonging to topics). Therefore, we proceeded to convert values of the features into binary values. To that end, a threshold was calculated dividing the sum of all probabilities of belonging by the number of topics established. Each probability that is equal to or greater than the threshold was converted into one; otherwise it was converted into zero.
- LIWC generates vectors of 64 features. The means of obtaining each vector was as follows. Given a document and the 64 categories, if some word of current category was found in the document, then such feature had the value of one, otherwise it had the value of zero.

Details on the WSM, LDA and LIWC will be given in Sect. 3.3. Classification was conducted by using a support vector network with fivefold cross-validation. We experimented with the following kernels:

- linear: $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$.
- polynomial: $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + r)^d, \gamma > 0$.
- radial basis function (RBF): $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \parallel \mathbf{x}_i - \mathbf{x}_j \parallel^2), \gamma > 0$.
- sigmoid: $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i^T \mathbf{x}_j + r)$.

We obtained best results (values between 59.8 and 76.3%) with the linear kernel (measured for the mixed-domain corpus). For polynomial, we obtained a classification of 50% in all corpora. For radial basis function, values between 49.5 and 52.5%, and finally, for a sigmoid kernel, we obtained values between 49.7 and 51.2% for all corpora. Since these values are near 50%, which corresponds to a random baseline, hereafter all results will be reported for SVN with linear kernel.

3.2 Datasets used

We experimented with three datasets: the DeRev corpus, the OpSpam corpus, and a corpus of opinions about three controversial topics. The authors of the corpora used two traditional methods to collect deceptive and truthful texts: sanctioned and unsanctioned deception (Gokhman et al. 2012).

DeRev dataset (DEception in REViews) (Fornaciari and Poesio 2014) is a corpus composed of deceptive and truthful opinions obtained from the Amazon Web site. This corpus includes opinions about books. This gold-standard corpus contains 236 texts, of which 118 are truthful and 118 are deceptive.

The confidence in that the deceptive texts were collected correctly is based on two communications, by Sandra Parker[1] and by David Streitfeld.[2] Parker claimed that she received a payment for writing opinions about 22 books. Streitfeld made known four books in which their authors admitted to be paid for writing opinions. DeRev's authors analyzed these communications and focused on twenty writers of fake opinions, which resulted in a corpus of 96 deceptive opinions.

To obtain the 118 truthful texts, DeRev's authors took into account certain aspects to ensure a high probability of that the selection was correct by making sure for the texts not to exhibit any cue of deception. Those cues of deception refer mainly on such aspects as whether opinions were written by users who used their real name, whether opinions were written by users who actually bought the book in question from Amazon, among others.

In this dataset, the deceptive and truthful texts were not obtained in a deliberate manner, i.e., the participants were not asked to write lies; instead, the texts were obtained after the participant has lied. Thus, this is a corpus of unsanctioned deception.

A sample of deceptive text

> I definitely fell in love with this book! I am a huge poetry fanatic. In all honesty, my initial thoughts of the title of this book were an instant reminder of the movie Final Destination, but of course, I was not expecting the book to be a depiction of the movie. In opposition, The Final Destination is a book filled with poetry of personal inner thoughts, struggles, and even pain. As a poet as well, it can be quite hard to explain your most intimate thoughts throughout a poetry piece. I can relate to every poem because of the situations of unanswered prayers, feeling blind throughout life, happiness, loneness, failure, not fitting in society, and so much more. Excellent book!

A sample of truthful text

> This is an enjoyable account of a man who wanted to live a time of solitude, so he built a cabin and lived by himself, thinking and writing down his thoughts. This is a good account of 19th century life, close to nature. This is probably one of those books every well-educated American should read. This was my first kindle purchase. I have been meaning to read "Walden" for years now and never got around to reading it until I obtained my kindle. First of all, I love the kindle for the variety

[1] http://www.moneytalksnews.com/3-tips-for-spotting-fake-product-reviews-%E2%80%93-from-someone-who-wrote-them/.

[2] http://www.nytimes.com/2011/08/20/technology/finding-fake-reviews-online.html?_r=1.

> of classic literature that is available. I do not live close to a public library, so having books delivered to my kindle is great!

OpSpam dataset (Opinion SPAM) (Ott et al. 2011) is a corpus composed of fake and genuine opinions about different hotels. It was collected from the Amazon Web site as well.

OpSpam's authors used the Amazon Mechanical Turk (AMT)[3] to generate deceptive opinions. Each participant was given the name of a hotel and its respective Web site; this information allowed writing a review of this hotel. The authors asked the participants to imagine that they worked in a hotel and the administrator asked them to write an opinion about the hotel, as if they were guests. The opinion was to appear real and highlight the positive aspects of the hotel. OpSpam's authors limited the submissions to one by participant, not allowing the same person to write more than one opinion. Additionally, opinions were restricted to those who lived in the USA and had an AMT approval rating of at least 90%. The participants had a maximum of thirty minutes to write the opinion and they were paid one dollar per accepted opinion. In this way, 400 deceptive texts were collected.

On the other hand, truthful opinions were collected from TripAdvisor.[4] First, 6977 opinions on the twenty most popular hotels were extracted. The authors eliminated 3130 opinions that did not have five stars, 41 ones that were not written in English, 75 ones that had less than 150 characters, and 1607 ones that were written by those who wrote for their first time on TripAdvisor. Eventually, 400 of the remaining texts were selected.

With this, OpSpam's authors collected a dataset composed of 800 texts in total. The participants were asked to write lies to obtain the deceptive text. As a result, this is a corpus of sanctioned deception.

A sample of deceptive text

> The Hyatt Regency Chicago hotel is perfectly located in the center of downtown Chicago. Whether you are going there for business or pleasure, it is in the perfect place. The rooms are large and beautiful and the ball room took my breath away. The wi-fi connection was perfect for the work I needed to do and the show at the Navy Pier was perfect for when I needed a break. Other hotels have nothing on the Hyatt. I just wish there was a Hyatt Regency in every city for all of my business trips.

A sample of truthful text

> I stayed for four nights while attending a conference. The hotel is in a great spot—easy walk to Michigan Ave shopping or Rush St., but just off the busy streets. The room I had was spacious and very well appointed. The staff was friendly, and the fitness center, while not huge, was well equipped and clean. I have stayed at a number of hotels in Chicago, and this one is my favorite. Internet was not free, but at $10 for 24 hours is cheaper than most business hotels, and it worked very well.

Opinions dataset (Pérez-Rosas and Mihalcea 2014a) is a corpus composed of opinions about three controversial topics: abortion, death penalty, and a best friend. It consists of 100 deceptive texts and 100 truthful texts. The collection of texts was conducted through AMT and the task for English originating from the USA was restricted to the participants who lived in the that country (the dataset also contains texts in English of speakers from India and Spanish of speakers from Mexico; however, these texts were not used in our research).

To obtain truthful texts, the authors asked participants to write a real opinion about each of the topics. In contrast, to obtain deceptive texts, the participants were asked to lie about their opinion; thus, this is a corpus of sanctioned deception.

A sample of deceptive text

> Abortion is murder and people who kill others should be put to death. It goes against the teachings of the Bible and is the worst kind of sin. We should do everything to stop it, no matter the cost. People should be ashamed for even thinking about having an abortion.

A sample of truthful text

> I believe that abortion in some cases is positive thing. Of course, in the case of rape or if the baby would be deformed or have no life quality it is acceptable. I do not believe abortion should be used as a form of birth control, meaning that every "mistaken" pregnancy can be dealt with an abortion. In some cases, extreme financial distress, perhaps it can be an alternative.

3.3 Sources of text features

We focused on three feature sources: two types of feature usually found in text deception identification works, namely word-space model (WSM) and the LIWC dictionary, and the semantic continuous space model (LDA).

Unigrams have been used in several works (see Sect. 2) as basis for combining with other kinds of features. This is because features based on unigrams are very informative in the task of deception detection. In addition, those features were commonly combined with LIWC for obtaining behavioral information latent in documents. However, the main drawback is that LIWC is a handcrafted resource; thus, a specific LIWC tool is necessary to analyze a different language.

We use a binary word-space model instead of unigrams due to the fact that both methods showed similar performance even when the former is simpler. Furthermore, we use

[3] http://www.mturk.com.

[4] http://www.tripadvisor.com.

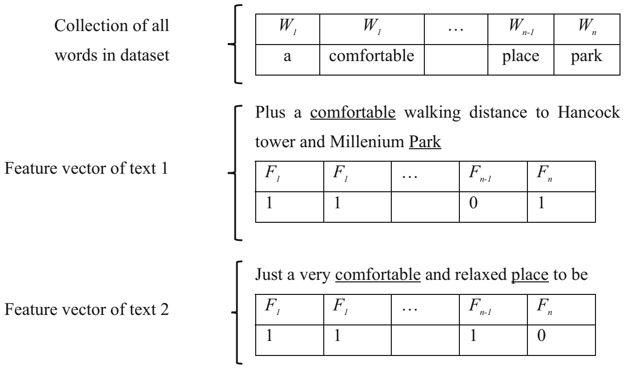
Collection of all words in dataset
W_1
W_1
…
W_n-1
W_n
a
comfortable
place
park
Plus a comfortable walking distance to Hancock tower and Millenium Park
Feature vector of text 1
F_1
F_1
…
F_n-1
F_n
1
1
0
1
Just a very comfortable and relaxed place to be
Feature vector of text 2
F_1
F_1
…
F_n-1
F_n
1
1
1
0

Fig. 1 This example shows how vectors of features are formed by using a binary word-space model

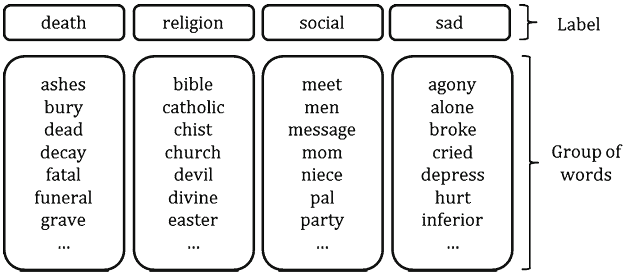

Fig. 2 Example of some groups of words with its corresponding label contained in LIWC (We show just a few of the words per group than we can find in LIWC)

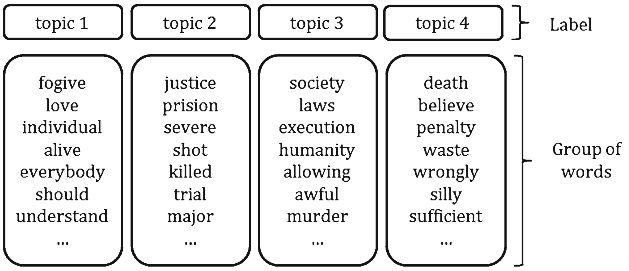

Fig. 3 Example of generated topics by using LDA in texts about death penalty

LDA for adding semantic information to the model. Unlike LIWC, LDA analyzes statistically documents to extract features regardless of the language in question.

Word-space model (WSM) Several previous works have shown that words are very important and relevant features for the task of identifying deceptive texts. For this reason, we decided to analyze the performance of features based on a matrix of words, which represent a word-space model.

To generate these features, we formed a list of all words $W_1, W_2, \ldots, W_n$ in the dataset. Then, we analyzed each document by searching whether W_n exists in the current text, in which case the feature $n(F_n)$ was set to 1, otherwise it was set to 0. Figure 1 shows how vectors of features were represented.

Linguistic inquiry and word count (LIWC) is a word counting tool (Pennebaker et al. 2007). It is based on groups of words manually labeled. LIWC classifies words in emotional, cognitive and structural component categories. It was developed for studying the psycholinguistic concerns dealing with the therapeutic effect of verbally expressing emotional experiences and memories. LIWC provides an English dictionary composed by nearly 4500 words and word stems. Each word can be classified into one or more of 64 categories. These are classified in four groups: linguistic processes (pronouns, articles, prepositions, numbers, negations), psychological processes (affective words, positive, negative emotions, cognitive process, perceptual process), relativity (time, space, motion), and personal concerns (occupation, leisure activity, money/financial issues, religion, death, and dying).

We used the version 2007 of LIWC with the English Dictionary as of 04/11/2013. Figure 2 shows an example of some groups of words found in LIWC.

Latent Dirichlet allocation (LDA) (Blei et al. 2003) is a probabilistic generative model for discrete data collections such as texts collection. It represents documents as a mix of different *topics*. Each topic consists of a set of words that keep some link between them. Words, in its turn, can be chosen based on probability. The model assumes that each document is formed word-by-word by randomly selecting a topic and a word for this topic. As a result, each document can combine different topics. Namely, simplifying things somewhat, the generation process assumed by the LDA consists of the following steps:

1. Determine the number N of words in the document according to the Poisson distribution.
2. Choose a mix of topics for the document according to Dirichlet distribution, out of a fix set of K topics.
3. Generate each word in the document as follows:
   (a) choose a topic;
   (b) choose a word in this topic.

Assuming this generative model, LDA analyzes the set of documents to reverse engineering this process by finding the most likely set of topics of which a document may consist. Unlike LIWC, LDA generates the groups of words (topics) automatically; see Fig. 3.

Accordingly, LDA can infer, given a fixed number of topics, how likely is that each topic (set of words) appear in a specific document of a collection. For example, in a collection of documents and 500 latent topics generated with the LDA algorithm, each document would have different distributions of 500 likely topics. That also means that vectors of 500 features would be created.

Table 1 Accuracy comparison with regard to number of established topics

Topics	Average accuracy (%)
100	74.77
200	79.54
300	83.68
400	86.24
500	87.86
600	**88.49**
700	88.19
800	88.13
900	88.04
1000	87.65

LDA requires the number of topics to generate to be specified; any change in this parameter may change the classification accuracy. For this reason, it is necessary to find an appropriate value. To find the number of topics that allows an optimal classification, we tested different values for LDA + WSM features on each corpus and calculated the average accuracy. Results of those experiments are shown in Table 1; In this table, the number of topics is compared against the obtained accuracy. In addition, it can be seen that by increasing the number of topics, it is possible to reach an optimal point from which increasing the number of topics does not imply an increase in accuracy (i.e., 600 topics).

All experiments hereafter involving LDA consider 600 topics. Each document processed by LDA generates a vector of 600 features, each one representing the probability that the document belongs to each topic. Note that all features are converted into binary values, as we detailed in Sect. 3.1, before classification.

Once that feature vectors are generated using different combinations of features, we proceed to train and test our model on different corpora. This allows us to answer our motivating questions: (1) To assess the appropriateness of support vector networks (SVN) for the task of classification of deceptive texts as precisely as possible, and (2) To explore whether a feature set can be sufficiently general to be used for classifying a dataset on a topic different from the topic of the dataset used for training.

4 Results and discussion

In this section, we present the results obtained on different combinations of datasets using an SVN. First, we performed our classification separately on each dataset, using fivefold cross-validation (for this, testing and training parts of each individual dataset were merged together) (Table 2). This experiment had better performance on most datasets (4 out of 5) with regard to other studies (see Sect. 4.1, Table 3); therefore, we decided to find out the scope of the proposed approach. For that reason, we experimented with mixed-domain classification on a dataset obtained by merging all datasets, with fivefold cross-validation. Finally, we experimented with cross-domain classification by using a concatenation of all-but-one corpora for training, with evaluation on the remaining dataset. We experimented with different kinds of feature and combinations of features: LDA, LIWC, and WSM, as described above.

4.1 In-domain results

First, we performed experiments on individual datasets; each one devoted to a specific subject domain, with fivefold cross-validation. Figures 4, 5 and 6 show the results on OpSpam, DeRev, and the controversial topics corpora, respectively. A combination of LDA and WSM features yielded the best results in all three cases.

Before proceeding to other experiments, such as mixed-domain and cross-domain deception detection, we wanted to make sure that our in-domain classifiers were performing

Table 2 Statistical significance

Dataset	# docs	This work (%)	Other Works	p-value	S. significance
OpSpam	800	90.9	Ott et al. (2011) (89.9%)	0.248	No
			Feng et al. (2012) 2012 (91.2%)	0.416	No
			Donato et al. 2015 (90.2%)	0.316	No
DeRev	236	94.9	Fornaciari and Poesio (2014) (76.3%)	0.000	Yes
Abortion	200	87.5	Pérez-Rosas and Mihalcea (2014a) (80.3%)	0.025	Yes
			Feng et al. (2012) (77.0%)	0.003	Yes
Best friend	200	87.0	Pérez-Rosas and Mihalcea (2014a) (75.9%)	0.002	Yes
			Feng et al. (2012) (85.0%)	0.283	No
Death penalty	200	80.0	Pérez-Rosas and Mihalcea (2014a) (77.2%)	0.248	No
			Feng et al. (2012) (71.5%)	0.023	Yes

Table 3 Comparison of our results with other works on the same corpora

Corpus	Works	Accuracy (%)
OpSpam	This work (LDA + WSM)	90.9
	Ott et al. (2011) (LIWC + bigrams)	89.8
	Feng et al. (2012) (syntactic rel. + unigrams)	**91.2**
	Donato et al. 2015	90.2
DeRev	This work (LDA + WSM)	**94.9**
	Fornaciari and Poesio (2014)	76.27
Abortion	This work (LDA + WSM)	**87.5**
	Pérez-Rosas and Mihalcea (2014a)	80.3
	Feng et al. (2012) (syntactic rel. + unigrams)	77.0
Best Friend	This work (LDA + WSM)	**87.0**
	Pérez-Rosas and Mihalcea (2014a)	75.9
	Feng et al. (2012) (syntactic rel. + unigrams)	85.0
Death Penalty	This work (LDA + WSM)	**80.0**
	Pérez-Rosas and Mihalcea (2014a)	77.2
	Feng et al. (2012) (syntactic rel. + unigrams)	71.5

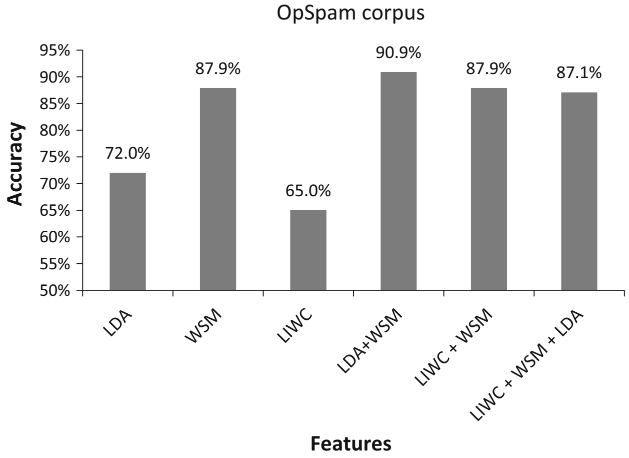

Fig. 4 Accuracy obtained on OpSpam corpus with different features

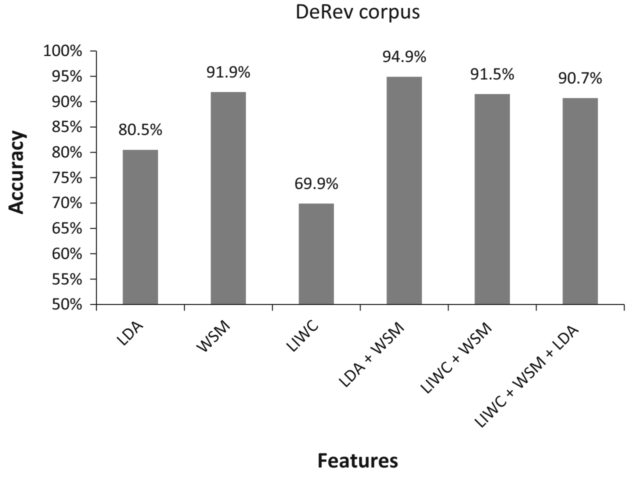

Fig. 5 Accuracy obtained on DeRev corpus with different features

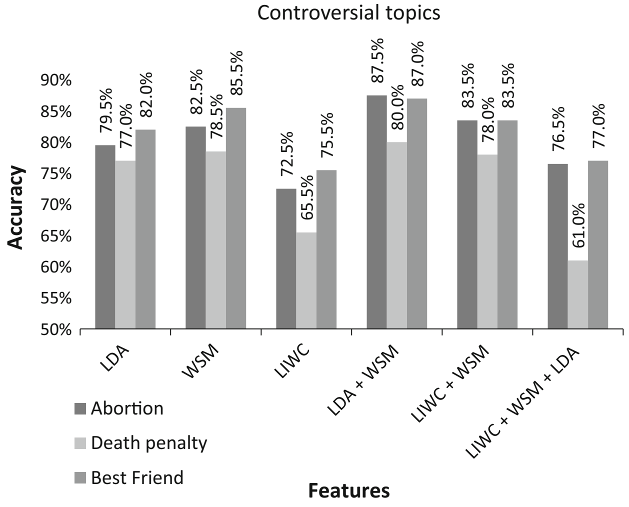

Fig. 6 Accuracy obtained on controversial topics corpus classification with different features

accordingly to the state of the art. Thus, we present a comparison of our results with other works in Table 3. Except for OpSpam, we performed better than other works currently known to us.

We show in Table 2 the statistical significance between this research results and other authors' results. For comparison purposes, we set a significance level (α) of 0.05 (5%), which means that statistical significance is attained if p-value is less than α. With this significance level, the improvement shown by some of our results would present no statistical significance when compared with other existing methods, making them practically equivalent; however, our approach has the main advantage that, unlike LIWC, LDA can be applied to different languages without needing a new tool for each lan-

Table 4 Accuracy, precision (P), recall (R) and F-measure (F) obtained on the combined corpus using SVN

Method	Accuracy	Truthful			Deceptive		
		P	R	F	P	R	F
LDA	65.8	67.6	60.9	64.1	64.4	70.8	67.4
WSM	73.5	73.6	73.5	73.5	73.5	73.6	73.5
LDA + WSM	**76.3**	**76.8**	75.6	**76.2**	**75.9**	**77.1**	**76.5**
LIWC	59.8	60.1	58.4	59.2	59.5	61.1	60.3
LIWC + WSM	73.7	71.9	**77.8**	74.7	75.8	69.7	72.6
LDA + LIWC	69.7	69.3	70.8	70.1	70.2	68.7	69.4
LIWC + WS + LDA	72.6	72.3	73.5	72.9	73.0	71.9	72.5

SVN outperformed other classifiers, such as Naïve Bayes; see Table 5

Table 5 Accuracy, precision (P), recall (R) and F-measure (F) obtained on the combined corpus using Naïve Bayes

Method	Accuracy	Truthful			Deceptive		
		P	R	F	P	R	F
LDA	61.1	61.5	59.3	60.4	60.7	62.8	61.7
WSM	**74.3**	73.9	75.3	**74.6**	74.8	73.3	74.1
LDA + WSM	73.5	71.9	**77.0**	74.4	**75.3**	69.9	72.5
LIWC	56.4	55.6	64.7	59.8	57.7	48.3	52.6
LIWC + WSM	73.3	73.3	73.3	73.3	73.3	73.3	73.3
LDA + LIWC	64.8	65.1	64.2	64.6	64.7	65.5	65.1
LIWC + WS + LDA	73.3	**76.0**	68.3	71.9	71.2	**78.4**	**74.6**

guage. Additionally, compared with the remaining works, our results present a statistically significant improvement.

4.2 Mixed-domain classification

The main aim of these experiments shown below was to investigate to what extent the SVN classifier can be used when we combine the five datasets on different domains, with fivefold cross-validation. With this, the training set contained subject domains (but not specific texts) that were also contained in the test set. In this case, again a combination of LDA and WSM features yielded the best result; see Table 4.

4.3 Cross-domain classification

Unlike the classification combining all domains, for this experiment we selected each dataset once as testing set and used the other remaining datasets as one combined training set. In this way, the subject domain of the test set was not included in the training set. Table 6 shows the results of cross-domain classification. In these experiments, unlike the experiments presented in Sects. 4.1 and 0, the combination of LDA and WSM features did not consistently yield the best accuracy. We show in boldface the best accuracy obtained for each dataset. In most cases (3 out of 5), the SVN outperformed NB; however, with a relatively simple setup of a plain word-space model, NB is able to improve deception detection with features learned from other datasets.

5 Conclusions and future work

Our motivation was, first, to assess the appropriateness of support vector networks (SVN) for the task of classification of deceptive texts, and, second, to explore whether a feature set can be sufficiently general to be used for classifying a dataset on a topic different from the topic of the dataset used for training, which would allow creating domain-independent general-purpose deception text detectors. For the first point, we can conclude that SVNs are indeed suitable and give good performance on deceptive text classification. We obtained the best results with a linear kernel.

As to the second question, we conducted two tests: the first one consisting on a mixed corpus, where the information of all training parts was merged to form a single combined corpus. For this test, we obtained an accuracy of 76.3% using SVN with LDA + WSM as features. This would be the expected performance if we had to detect deception within one of the subjects covered in the datasets, i.e., opinions about books, hotels, best friends, abortion, and death penalty.

The second test consisted in detecting deception in a domain that had not been seen at all in training. This would allow us to measure the degree of prediction that could be achieved in order to classify a text that was not in the scope of the datasets. In this test, we slightly surpassed the baseline of random classification given two classes (deceptive vs. truthful), with results ranging from 53.8% on the OpSpam corpus to 64% on the "best friend" corpus. In most cases, though not

Table 6 Accuracy obtained in cross-domain classification

		DeRev	OpSpam	Abortion	Best friend	Death penalty	Average
LDA	SVN	52.1	48.8	57.5	50.0	53.5	50.43
	NB	43.2	49.8	56.0	51.5	56.0	46.50
WSM	**SVN**	53.3	52.8	55.5	55.5	**58.5**	53.05
	NB	50.8	**53.8**	**58.5**	56.0	48.5	52.30
LIWC	SVN	51.3	49.2	53.1	54.3	52.8	50.25
	NB	50.9	51.1	52.6	56.8	51.4	51.00
LIWC + WSM	SVN	54.6	**53.8**	54.5	55.0	56.0	54.20
	NB	47.8	52.5	55.5	59.0	51.0	50.15
LDA + WSM	**SVN**	**59.3**	50.6	57.5	**64.0**	55.0	54.95
	NB	58.8	52.3	55.0	59.5	52.5	**55.55**
LDA + LIWC	SVN	56.3	46.3	54.5	55.5	52.0	51.30
	NB	45.7	48.1	46.5	51.5	49.5	46.90
LDA + LIWC + WSM	SVN	52.1	52.6	57.0	57.0	54.0	52.35
	NB	52.9	53.0	58.0	62.5	53.0	52.95
	Best	SVN	NB/SVN	NB	SVN	SVN	NB

always, the best classification result was obtained by SVN with LDA + WSM features. For some datasets (OpSpam and Abortion), the NB classifier with WSM features alone performed better than SVN. On average, NB with LDA + WSM features provided the best results (55.55%).

In general, LDA-based features provide a means for generalizing deception cues in terms of semantic topics; however, this seems to yield only a slight increase in performance in terms of a more general deception detector.

As a future work, we plan to investigate other features, such as syntactic style patterns, among others, which could help to identify deception in a broader purpose range.

Acknowledgements We thank Instituto Politécnico Nacional (SIP, COFAA and BEIFI), and SNI. Partially funded by CONACyT (Language Technologies Thematic Network Projects 260178, 271622) and SIP Project Number 20162058

Compliance with ethical standards

Conflict of interest Authors declare they have no conflict of interest.

Human and animal rights This article does not contain any studies with human participants performed by any of the authors.

References

Almela A, Valencia-García R, Cantos P (2012) Seeing through deception: a computational approach to deceit detection in written communication. In: Proceedings of the EACL 2012 workshop on computational approaches to deception detection. Avignon, France, pp 15–22

Al-Shammari ET, Keivani A, Shamshirband S, Mostafaeipour A, Yee L, Petković D, Ch S (2016) Prediction of heat load in district heating systems by support vector machine with firefly searching algorithm. Energy 95:266–273

Al-Shammari ET, Mohammadi K, Keivani A, Ab Hamid SH, Akib S, Shamshirband S, Petković D (2016b) Prediction of daily dewpoint temperature using a model combining the support vector machine with firefly algorithm. J Irrig Drain Eng 142(5):04016013

Altameem TA, Nikolić V, Shamshirband S, Petković D, Javidnia H, Kiah MLM, Gani A (2015) Potential of support vector regression for optimization of lens system. Comput Aided Des 62:57–63

Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022

DePaulo BM, Lindsay JJ, Malone BE, Muhlenbruck L, Charton K, Cooper H (2003) Cues to deception. Psychol Bull 129:74–118

Ekman P (1989) Why lies fail and what behaviors betray a lie. In: Yuille JC (ed) Credibility assessment. Kluwer, New York, pp 71–81

Feng S, Banerjee R, Choi Y (2012) Syntactic stylometry for deception detection. In: Proceedings of the 50th annual meeting of the association for computational linguistics, Republic of Korea, Jeju, pp 171–175

Fornaciari T, Poesio M (2014) Identifying fake amazon reviews as learning from crowds. In: Proceedings of the 14th conference of the European chapter of the association for computational linguistics, Gothenburg, Sweden, pp 279–287

Gani A, Mohammadi K, Shamshirband S, Altameem TA, Petković D, Ch S (2016) A combined method to estimate wind speed distribution based on integrating the support vector machine with firefly algorithm. Environ Prog Sustain Energ 35: 867–875. doi:10.1002/ep.12262

Gocic M, Shamshirband S, Razak Z, Petković D, Ch S, Trajkovic S (2016) Long-term precipitation analysis and estimation of precipitation concentration index using three support vector machine methods. Adv Meteorol 2016:7912357. doi:10.1155/2016/7912357

Gokhman S, Hancock J, Prabhu P, Ott M, Cardie C (2012) In search of a gold standard in studies of deception. In: Proceedings of the EACL 2012 workshop on computational approaches to deception detection, Avignon, France, pp 23–30

Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182

Hall MA (1999) Correlation-based feature selection for machine learning. Ph.D. Thesis. Department of Computer Science, University of Waikato

Hall MA, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The WEKA data mining software: an update. SIGKDD Explor Newsl 11(1):10–18

Hauch V, Blandón-Gitlin I, Masip J, Sporer SL (2012) Linguistic cues to deception assessed by computer programs: a meta-analysis. In: Proceedings of the EACL 2012 workshop on computational approaches to deception detection, Avignon, France. Association for Computational Linguistics, pp 1–4

Hernández Fusilier D, Montes-y-Gómez M, Rosso P, Guzmán Cabrera R (2015) Detection of opinion spam with character n-grams. In: Proceedings of 16th international conference on intelligent text processing and computational linguistics, Cairo, Egypt, pp 285–294

Jović S, Danesh AS, Younesi E, Aničić O, Petković D, Shamshirband S (2016a) Forecasting of underactuated robotic finger contact forces by support vector regression methodology. Int J Pattern Recognit Artif Intell 30(7). doi:10.1142/S0218001416590199

Jović S, Radović A, Šarkoćević Ž, Petković D, Alizamir M (2016b) Estimation of the laser cutting operating cost by support vector regression methodology. Appl Phys A 122(9):798

Keila PS, Skillicorn DB (2005) Detecting unusual email communication. In: CASCON 2005, pp 117–125

Kisi O, Shiri J, Karimi S, Shamshirband S, Motamedi S, Petković D, Hashim R (2015) A survey of water level fluctuation predicting in Urmia Lake using support vector machine with firefly algorithm. Appl Math Comput 270:731–743

Mihalcea R, Strapparava C (2009) The lie detector: explorations in the automatic recognition of deceptive language. In: Proceedings of the ACL-IJCNLP, ACL-IJCNLP. Suntec, Singapore, pp 309–312

Mohammadi K, Shamshirband S, Anisi MH, Alam KA, Petković D (2015) Support vector regression based prediction of global solar radiation on a horizontal surface. Energy Convers Manag 91:433–441

Mohammadi K, Shamshirband S, Tong CW, Arif M, Petković D, Ch S (2015) A new hybrid support vector machine–wavelet transform approach for estimation of horizontal global solar radiation. Energy Convers Manag 92:162–171

Newman ML, Pennebaker JW, Berry DS, Richards JM (2003) Lying words: predicting deception from linguistic styles. Personal Soc Psychol Bull 29(5):665

Olatomiwa L, Mekhilef S, Shamshirband S, Petkovic D (2015) Potential of support vector regression for solar radiation prediction in Nigeria. Nat Hazards 77(2):1055–1068

Olatomiwa L, Mekhilef S, Shamshirband S, Mohammadi K, Petković D, Sudheer C (2015) A support vector machine–firefly algorithm-based model for global solar radiation prediction. Solar Energy 115:632–644

Ott M, Choi Y, Cardie C, Hancock JT (2011) Finding deceptive opinion spam by any stretch of the imagination. In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, Portland, Oregon, pp 309–319

Pennebaker JW, Chung CK, Ireland M, Gonzales A, Booth RJ (2007) The development and psychometric properties of liwc2007. LIWC.Net, Austin

Pérez-Rosas V, Mihalcea R (2014a) Cross-cultural deception detection. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, pp 440–445

Pérez-Rosas V, Mihalcea R (2014b) Gender differences in deceivers writing style. 2014. In: 13th Mexican international conference on artificial intelligence. Tuxtla Gutiérrez, Chiapas, México, pp 163–174

Petković D, Shamshirband S, Saboohi H, Ang TF, Anuar NB, Pavlović ND (2014) Support vector regression methodology for prediction of input displacement of adaptive compliant robotic gripper. Appl Intell 41(3):887–896

Petković D, Shamshirband S, Saboohi H, Ang TF, Anuar NB, Rahman ZA, Pavlović NT (2014b) Evaluation of modulation transfer function of optical lens system by support vector regression methodologies–a comparative study. Infrared Phys Technol 65:94–102

Piri J, Shamshirband S, Petković D, Tong CW, ur Rehman MH (2015) Prediction of the solar radiation on the Earth using support vector regression technique. Infrared Phys Technol 68:179–185

Protić M, Shamshirband S, Petković D et al (2015) Forecasting of consumers heat load in district heating systems using the support vector machine with a discrete wavelet transform algorithm. Energy 87:343–351

Schelleman-Offermans K, Merckelbach H (2010) Fantasy proneness as a confounder of verbal lie detection tools. J Investig Psychol Offender Profiling 7:247–260

Shamshirband S, Mohammadi K, Khorasanizadeh H, Yee L, Lee M, Petković D, Zalnezhad E (2016) Estimating the diffuse solar radiation using a coupled support vector machine–wavelet transform model. Renew Sustain Energy Rev 56:428–435

Shamshirband S, Petković D, Amini A et al (2014) Support vector regression methodology for wind turbine reaction torque prediction with power-split hydrostatic continuous variable transmission. Energy 67:623–630

Shamshirband S, Petković D, Javidnia H, Gani A (2015) Sensor data fusion by support vector regression methodology—a comparative study. IEEE Sens J 15(2):850–854

Shamshirband S, Petković D, Pavlović NT, Ch S, Altameem TA, Gani A (2015) Support vector machine firefly algorithm based optimization of lens system. Appl Opt 54(1):37–45

Shamshirband S, Tabatabaei M, Aghbashlo M, Yee L, Petković D (2016b) Support vector machine-based exergetic modelling of a DI diesel engine running on biodiesel-diesel blends containing expanded polystyrene. Appl Therm Eng 94:727–747

Shenify M, Danesh AS, Gocić M, Taher RS et al (2016) Precipitation estimation using support vector machine with discrete wavelet transform. Water Resour Manag 30(2):641–652

Toma CL, Hancock JT (2012) What lies beneath: the linguistic traces of deception in online dating profiles. J Commun 62:78–97

Twitchell DP, Nunamaker JF Jr, Burgoon JK (2004) Using speech act profiling for deception detection. In: Intelligence and security informatics: second symposium on intelligence and security informatics, ISI 2004, pp 403–410

Williams SM, Talwar V, Lindsay RCL, Bala N, Lee K (2014) Is the truth in your words? Distinguishing children's deceptive and truthful statements. J Criminol 2014:547519. doi:10.1155/2014/547519

Xu Q, Zhao H (2012) Using deep linguistic features for finding deceptive opinion spam. In: Proceedings of COOLING, pp 1341–1350