Knowledge-Based Systems 149 (2018) 155–168

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Applying automatic text-based detection of deceptive language to police reports: Extracting behavioral patterns from a multi-step classification model to understand how we lie to the police

Lara Quijano-Sánchez[a,*], Federico Liberatore[a,b], José Camacho-Collados[c], Miguel Camacho-Collados[d]

[a] UC3M-BS Institute of Financial Big Data, Universidad Carlos III de Madrid, Getafe, Madrid, Spain
[b] Department of Statistics and Operational Research, Universidad Complutense de Madrid, Getafe, Madrid, Spain
[c] Department of Computer Science, Università degli Studi di Roma "La Sapienza", Rome, Italy
[d] State Secretariat for Security, Interior Ministry, Madrid, Spain

ARTICLE INFO

Article history:
Received 30 October 2017
Revised 5 March 2018
Accepted 7 March 2018
Available online 8 March 2018

Keywords:
Lie detection
Information extraction
Predictive policing
Model knowledge extraction
Natural language processing
Decision support systems

ABSTRACT

Filing a false police report is a crime that has dire consequences on both the individual and the system. In fact, it may be charged as a misdemeanor or a felony. For the society, a false report results in the loss of police resources and contamination of police databases used to carry out investigations and assessing the risk of crime in a territory. In this research, we present *VeriPol*, a model for the detection of false robbery reports based solely on their text. This tool, developed in collaboration with the Spanish National Police, combines Natural Language Processing and Machine Learning methods in a decision support system that provides police officers the probability that a given report is false. *VeriPol* has been tested on more than 1000 reports from 2015 provided by the Spanish National Police. Empirical results show that it is extremely effective in discriminating between false and true reports with a success rate of more than 91%, improving by more than 15% the accuracy of expert police officers on the same dataset. The underlying classification model can be analysed to extract patterns and insights showing how people lie to the police (as well as how to get away with false reporting). In general, the more details provided in the report, the more likely it is to be honest. Finally, a pilot study carried out in June 2017 has demonstrated the usefulness of *VeriPol* on the field.

1. Introduction

A false report is a crime governed by federal and state laws. It involves a person who -with the intent to deceive- makes a false statement to a police or law enforcement officer, influencing the outcome of a criminal investigation. Filing a false police report can have very serious consequences on both the individual and the system. Depending on the country, it may be considered as a misdemeanor or a felony, charges which could result in jail terms and/or fines. Besides, in the case of robberies, it has been observed that a false police report is generally followed by other crimes, such as frauds, which result in even more serious charges. For society, a false report represents a waste of public resources that should be dedicated to pursuing other crimes. Also, each undetected false report contaminates police databases used to carry out investigations as well as assessing the risk of crime in a territory, which is then used to take strategic decisions, at both local and national level. Despite its seriousness, this type of crime is extremely common and it is generally carried out by citizens that do not have a previous criminal record. Fig. 1(a) shows the number of false robbery reports in Spain in 2015, by urban area. These numbers represent a lower bound to the real number of false reports as it is impossible to know with certainty their exact number. In fact, our data shows that more than 80% of the robbery cases are left unsolved. Fig. 1(b) illustrates the percentage of robbery reports identified as false in Spain in 2015, by urban area. As it can be seen, the efficacy in detecting false robbery reports is extremely heterogeneous, ranging from 0% to 20%. Based on the performance of the most successful Police Department in detecting false robbery reports in 2015, a rough estimation of the ratio of falsehood is 57% approx.

* Corresponding author.
E-mail addresses: laraquij@inst.uc3m.es (L. Quijano-Sánchez), fliberat@inst.uc3m.es (F. Liberatore), collados@di.uniroma1.it (J. Camacho-Collados), mcc@interior.es (M. Camacho-Collados).

https://doi.org/10.1016/j.knosys.2018.03.010
0950-7051/

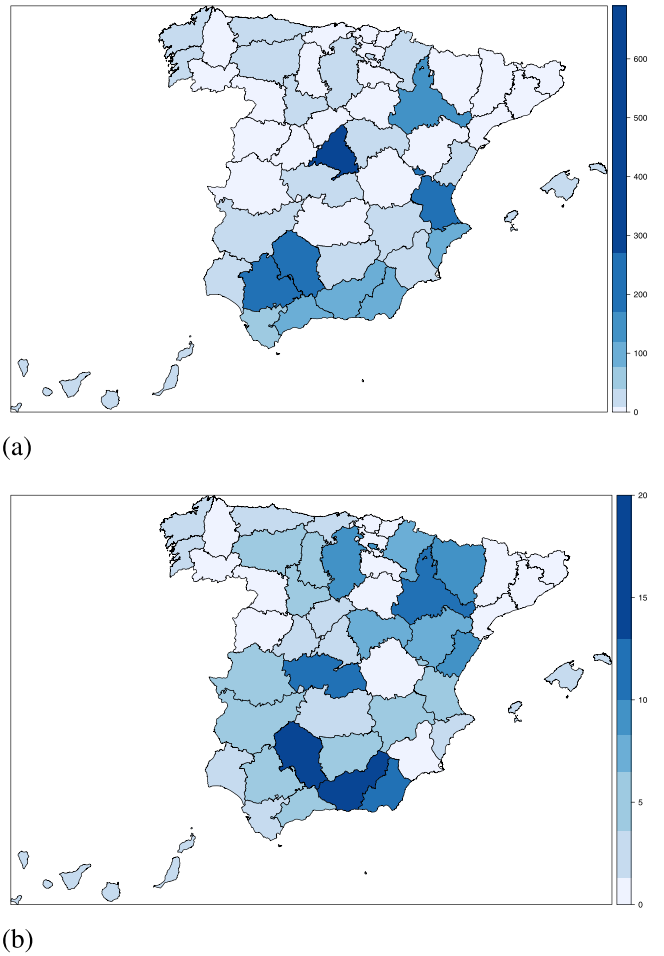

Fig. 1. (a) Number of false robbery reports in Spain in 2015, by urban area. (b) Percentage of robbery reports identified as false in Spain in 2015, by urban area. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Is it possible to differentiate between true and false robbery reports based exclusively on the wording of the report? This was the question put to us by the representatives of the Spanish National Police (SNP). They also argued that a tool with such capabilities would represent a revolution for police work, an advancement in terms of methodology, and a change of paradigm in the way that the police investigates. According to them, this system would help the police to better focus their resources and, when properly publicized, its mere existence would discourage citizens from filing a false report, thus preventing the commission of crimes. A win-win situation!

We tackled this problem within the framework of Data Science. The methodology used is a combination of Natural Language Processing (NLP) [1] algorithms, feature selection methods and Machine Learning (ML) [2] classification algorithms. The result is *VeriPol*, a model embedded in a decision support system capable of analyzing automatically the text of a robbery report and estimating its probability of falsehood with a high precision, empirically measured to be greater than 91%. *VeriPol* perfectly integrates with SIDENPOL, the Spanish National Police report information system. One of the most interesting features of this tool is the possibility to analyze its structure to extract knowledge and identify the most relevant differences between true and false robbery reports. This allows us to understand how people lie to the police, as well as how a true report looks like. In short, we could use *VeriPol* to learn how people lie to the police. Finally, it should be noted that it is the first tool of this type and represents a real novelty at both police and academic level. In fact, the research in lie detection on text is taking its first steps and *VeriPol* is the first model that has been estimated and validated on real documents, unlike current contributions in the literature using fictitious texts, written specifically for research [3,4]. Having a lie detection model that is able to automatically predict the truthfulness of a report is a key aspect in the usability and applicability of the tool in comparison to other predictive policing methods that need to be manually fed with key features in order to make a prediction [5,6].

In summary, we are dealing with a task of detection and classification of texts that use verbal deception. This field, although currently a hot topic [7–9], is quite new which makes it difficult to compare our contributions against others concerned with the detection of false or deceptive texts [3,4,10,11]. These works, that are mostly centered on the domain of spam detection and fake reviews [12–15], make use of traditional ML techniques such as Naive Bayes, Support Vector Machine (SVM), logistic regression or decision trees [16,17]. A comparison with several datasets on different domains from the literature shows that *VeriPol* outperforms all the considered previous models.

Consequently, this work has the following main contributions:

- Proposes a novel model for lie detection in written text that incorporates heuristic and optimal feature selection criteria and outperforms previous models from the literature.
- Introduces *VeriPol*, an innovative tool in the predictive policing domain that automatically extracts attributes fed to the predictive model against manually fed artificial tools in the same domain [5,6].
- Presents a case study on real documents (instead of on an artificial ad-hoc corpus as in other lie detection contributions [3,4]) in the novel application context of false report detection. The case study includes a pilot study that shows the capabilities of the proposed methodology on the field.
- Illustrates the hidden patterns and characteristics, obtained by a model-driven analysis of *VeriPol*, that can be used to differentiate between true and false robbery reports.

The reminder of the paper is organized as follows. In the next section we present the state of the art on predictive policing and detection of deceptive texts. In Section 3 the design of our lie detection model is illustrated in detail. Next, Section 4 presents computational experiments that show *VeriPol*'s performance on a real dataset, its improvement against state-of-the-art models, a comparison with human experts evaluations, and the resulting trained model. Next, in Section 5, the model is analyzed to extract patterns of falsehood and truthfulness in robbery reports. Section 6 shows the impact of *VeriPol* in the Spanish National Police by means of an on-the-field pilot study that took place in 10 Police Departments in June 2017. The article concludes with a summary of the main findings and some possible future lines of research.

2. Related work

In this section, a revision of the state-of-the-art on predictive policing models is presented along with the relevant literature on detection of false or deceptive statements.

2.1. State of the art on predictive policing

Up to date, in the specific domain of this research (i.e., false reporting or false confessions in policing) there is a limited number of studies dealing with the subject. Among the few existing, Gudjonsson et al. [18] statistically analyzed case characteristics of false confessions during interrogation. Also in this line, but on a much delicate subject, Lisak et al. [19] studied the actual low percentage

of false allegations of sexual assault as well as their characteristics. Ofshe and Leo [20] provided an analysis of USA's police interrogation techniques that relates their traits to the resulting type of confession (i.e., guilty or not). In Kane [21] a chi-square automatic interaction detector combined with logistic regression models was used to predict the probability of being arrested. Also, insights such as possible police biases identified by detected patterns are presented.

Regarding research in crime forecasting, a review on predictive policing practices can be found in [22] and [23]. It is important to note that, although the implications of crime forecasting are straightforward, the degree in which predictive policing actually prevents crime is still an open debate [24]. In [22], authors claim that currently there are four different types of predictive methods:

- Methods for predicting felonies: used to forecast places and times with crime escalation.
- Methods for predicting transgressors: used to identify individuals at risk of committing a felony in the future.
- Methods for predicting transgressors identities: used to shape profiles that precisely match likely transgressors with specific past felonies.
- Methods for predicting victims of felonies: used to identify groups, prototypes, or in some cases, individuals who are likely to become victims of a felony.

The research object presented in this paper does not fit in any of these categories. Therefore, it is necessary to define a new type of predictive method, namely: "Methods for predicting the veracity of felony victims statements." To the best of the authors' knowledge, *VeriPol* is the first contribution to this novel line of applications.

Within the framework of police operations support systems - yet in operational contexts different from *VeriPol*, such as patrolling [25,26] - there are several works that support tactical deployment of police resources through modeling. For example, Cohen et al. [5] created forecasting models of simple linear regressions and neural networks that predict with an accuracy of $R^2 = [0.690.79]$ whether a crime is violent or not based on a fixed set of 14 dependent variables chosen based on police requirements and data limitations. Furthermore, Yu et al. [6] presented an approach that consists of manually architecting datasets from original crime records and using data mining classification techniques [27] to predict crime "hotspots."

Reviewed predictive policing literature [5,6,25] relied on the manual introduction of data to train their models. Against it, *VeriPol* extracts prediction factors directly from the text of a report. Thus, the provided tool can be in fact used, installed and automatically recalibrated without human effort. It is important to notice that introducing data manually to feed a model is not a task that police officers would do, differently from an academic research setting. Therefore, having a lie detection model that is able to automatically predict the truthfulness of a report is a key aspect in the usability and applicability of the tool in comparison to other predictive policing methods that need to be manually fed with key features to make a prediction.

2.2. State of the art on deceptive texts

On a broader scope, within the framework of lie detection, failed attempts were made to introduce scans of brain activity for use in the courtroom as proof of lying [28]. However, very little work has been carried out on the automatic and non-invasive detection of deceptive language in written text. Most of the previous work has focused on the psychological or social aspects of lying [29,30]. In the field of NLP, the number of contributions concerned with the detection of false or deceiving texts is, again, very limited [3,4,10,11,31]. One of the first researches on the subject is the work of Zhou et al. [32], who analyzed linguistic cues for deception detection in the context of text-based asynchronous computer mediated communication. More recently, Mihalcea and Strapparava [33] presented initial experiments in the recognition of deceptive language based on three self-generated datasets that automatically classify texts as true or false with a 70% accuracy. Authors concluded that differentiating between truthful and lying texts is possible, as these classes present characteristics that make them separable.

Detecting automatically verbal deception [7], in court [34], in reviews [9,15,35] or in political debates [8] is a very hot topic in the research community. Most contributions in this line are mainly concerned with the domain of spam detection and fake reviews [12–14]. These works relate to this paper's topic as they reflect fictitious opinions that have been intentionally written to sound trustworthy. They mostly follow traditional ML techniques such as Naive Bayes, Support Vector Machine (SVM), logistic regression or decision trees [16,17]. Precision of the aforementioned research works is mostly comprised around the range of 70–90%. To the best of the authors' knowledge, to date the best designed technique in a similar domain is by Ott et al. [12] that accurately classify 89.8% deceptive hotel reviews by using a combination of bigrams, a psycholinguistic deception detection tool LIWC (Linguistic Inquiry and Word Count software [36]) and a SVM classifier. However, this papers' methodology exceeds current accuracy results regarding the automatic detection of verbal deception, as illustrated in Section 4.1, both in the same domain as Ott et al. [12] , and in other domains [9,14,33,37–39]. Despite of the domain difference (i.e., false reports VS fake reviews) and the nature of the document (i.e., facts collected and written by a policeman after questioning the victim of a felony VS a first person written text), the insights drawn by analyzing the structure of the estimated probability model seem to lead to common patterns also identified by other researchers in the fake reviews domain [3,4,12,40].

All in all, research on lie detection models is still at its infancy. More importantly there is no previous study on systems that are able to identify false police reports, neither on tools that make a prediction of the falsehood of a document by automatically analyzing its text.

3. Methodology

3.1. Context and data

VeriPol, our lie detection model, is trained using a corpus comprised of 1122 robbery reports filed in Spain in 2015. Each report represents a document in our corpus. The corpus includes 534 true reports and 588 false reports. All the reports have been anonymized by removing any personal information. Also, the reports only include the text of the declaration of the complainant, without any further information (e.g., location, date, time, other witnesses or agents reports). The procedure for the selection of the corpus is described in the following paragraph.

An officer with extensive experience in interrogation, lie detection, and investigation has been involved in the process of reviewing and classifying all the reports. This operation ensures the accuracy of the labeling process. In fact, thanks to this step, many reports whose classification was uncertain have been disregarded. Reviewing a report is a highly time consuming operation that requires a lot of experience and training. Therefore, the involvement of an expert in this phase is of primary importance and utterly necessary for the generation of a feasible corpus. The officer worked on this process over a two year period, after which the labeling process came to a halt. Reports were presented to the officer

in a random order, alternating potentially true and false reports. As a consequence, the sample is almost balanced. This was done because the real ratio of false report cannot be known, therefore the dataset was built in such a way that both classes are equally represented. Based on official statistics, only 3.98% of Spanish robbery reports registered in 2015 were proved to be false. Considering that 81.25% of the registered robbery cases have not been closed, the real percentage of false robbery reports could be much higher. In fact, the most successful Police Department in clearing false robbery reports in 2015 (154 proven false reports), experienced a ratio of false robbery reports of 57.68% in 2015. Assuming that the distribution of false reports is homogeneous across the country, this figure highlights that there is a lot of room for improvement.

3.2. Model design

Our objective is to estimate the probability of falsehood of a given document. We propose a model based on NLP preprocessing and ML techniques for probability estimation. The procedure proposed, consists of several steps explained in detail in the following.

Step 1: Text Preprocessing. In this first step the corpus is normalized by following a NLP preprocessing pipeline [41]: lowercasing, tokenization, lemmatization and stopword removal. The software TreeTagger [42] was used for tokenization and lemmatization. While tokenization and stopword removal are standard prior steps for normalizing texts, lowercasing and lemmatization are aimed at reducing sparsity and vocabulary size, steps which have been proved beneficial in text classification tasks [43–46]. As far as the stopword removal step is concerned, not only function words (e.g., articles and conjunctions) are removed, but also a tailored list of words for this specific task has been included to avoid certain biases into our model. Recent works have shown the need of reducing bias on supervised models, as the data they are trained on may contain undesirable biases [47–49]. In our particular case province and autonomous region names are filtered. The rationale is that the data showed that the ratio of false robbery reports detection was not homogeneous across Spain. However, according to experts, this is not due to differences in the behavior of the citizens, but rather on the skill of agents in detecting false reports or on the priority assigned in the Police Departments to the investigation of this type of crime.

Step 2: Feature Generation. Unigram lemma frequencies are used as features. Additionally, the number of tokens, lemmata (total and unique) and sentences within a document are also considered. Finally, we count the number of fine-grained Part-of-Speech (PoS) tags[1] per document and add them as features[2] See Table A.16 for the full list of PoS tags and their description. To obtain other measures that assess different magnitudes, beside frequencies, these features are further transformed using the following functions:

- Binary transformation, $bin(x)$: given feature x, $bin(x)$ takes value 0 if $x = 0$, 1 otherwise. This transformation measures the effect of the feature appearing in the document.
- Logarithm transformation, $log(x)$: natural logarithm of feature x. This transformation measures the effect of different orders of magnitude in the value of the features.
- Ratio transformation, $rat(x)$: all features are divided by the length of the document, except the unigram features that are divided by their total number of occurrences. This transformation measures the effect of proportions with respect to the total.

[1] PoS tagging is also performed by TreeTagger.

[2] The terms “feature” and “variable” are used interchangeably throughout this text.

Therefore, each document i is represented by vector x_i, including all the features and the transformations explained above. This vector is composed of $4 \cdot N_u + 4 \cdot 79$ dimensions, where N_u is the number of unigrams after the preprocessing and 79 represents the number of PoS tags plus the number of tokens, lemmata (total and unique) and sentences in the document.

Step 3: Feature Selection. To-date research in text classification has widely ignored feature selection techniques in their approaches [16], even when using text features, which tend to produce hugely dimensional feature sets. Mukherjee et al. [50] is a marked exception, as authors use Information Gain to perform feature selection of top 1 and 2%. Despite the low effect on the classifier performance, based on other domains' results, feature selection techniques can conceivably improve performance. However, this step should be crucial considering that one of the major difficulties in text classification task are data sparsity and high dimensionality of the feature space [51].

The unigram feature extraction procedure presented in Steps 1 and 2 produces a large number of variables. Among them, there could be unigram tokens with a low appearance rate that are present exclusively (or almost exclusively) in one class of document. These tokens, though very useful to classify, are not generalizable as they may only represent a small fraction of the documents, which in turn leads to overfitting and biased insights. To prevent this, a multi-step feature selection methodology is proposed, comprised of heuristic filters and optimization procedures, to exclude noisy or not general features.

First, all unigram features that comply with one of the following conditions are disregarded:

- Low in-sample appearance rate, i.e., less than or equal to 1% of the sample size.
- Very high appearance rate (i.e., greater than or equal to 99% of the sample size) but low variability, i.e., the coefficient of variation is less than or equal to the coefficient of variation of a sequence of numbers showing the minimum level of variability required. This sequence is built as a sample of numbers having the same length of the corpus and comprised exactly of: 1% of zeros, 0.5% of ones, 99% of twos, and 0.5% of threes, to represent a variable that shows very low variability in most of the sample.

Note that when a unigram is disregarded, the corresponding transformations (Binary, Logarithm and Ratio, see Step 2) are ignored too.

Second, a LASSO model with binomial link function is run on all the remaining variables (unigrams and document variables, and their transformations, see Step 2). LASSO (Least Absolute Shrinkage and Selection Operator) [52] is a regression analysis method that performs variable selection and, to a lesser extent, regularization to enhance the prediction accuracy and interpretability of the statistical model it produces. Lasso is able to achieve both of these goals by penalizing the sum of the absolute value of the regression coefficients in the optimization goal of the logistic model, which forces certain coefficients to be set to zero, effectively choosing a simpler model that does not include those coefficients. Depending on the value of the penalization coefficient different models can be obtained. In this procedure, 100 different penalization values are considered, according to the methodology presented by Friedman et al. [53]. The variables having non-zero regression coefficients in the most accurate model are selected while the others are disregarded.

A couple of improvements are still possible. In case of perfectly correlated variables, LASSO considers only one among them, arbitrarily. Therefore, it is possible that a more interpretable version

of the chosen variable could have been selected instead. In general, the preference order is the following: binary > frequency > logarithm > ratio. To simplify interpretation, after the LASSO procedure, all the selected variables are checked to see if a preferred version of the variable is perfectly correlated with the incumbent one. In this case, the incumbent variable is substituted with the most interpretable available. Note that this does not have an impact on the subsequent classification.

Finally, in case of groups of perfectly correlated variables in the dataset, LASSO could choose any number of them as it is indifferent with respect to the coefficient penalization part of its objective function. However, for the sake of interpretability, it would be better to include only one among perfectly correlated variables, possibly the most interpretable. Therefore, a procedure for the elimination of duplicated complex variables is implemented. Note that this procedure does not have an impact on the subsequent classification either.

Step 4: Probability Estimation with Ridge Logistic Regression. Ridge Regression [54] is a very commonly used method of regularization. This is achieved by penalizing the sum of the squared regression coefficients in the optimization goal of the logistic model, forcing the coefficients to take small values.

In this last step, multiple Ridge Logistic Regression models are estimated by varying the penalty value. Again, 100 different penalization values generated according to the methodology presented by Friedman et al. [53] are considered. The model having the best accuracy is then selected to estimate the probability of falsehood of reports.

4. Computational experiments

To validate the presented proposal as *VeriPol*'s predictive model, an empirical evaluation of the goodness of its performance against real tagged reports is needed. To do so, the dataset illustrated in Section 3.1 is used. This dataset, as previously described, is comprised of 588 false and 534 true reports, corresponding to a total of $n = 1,122$ cases. This number of documents is enough to overcome stated algorithmic classification problems due to small datasets [55].

VeriPol estimates the probability of falsehood of a report by using Ridge Logistic Regression. However, other approaches are possible and equally valid. Hence, next we present a comparison of regression methodologies, showing that Ridge Logistic Regression is the best choice. All in all, the following methods are tested:

1. Ridge Logistic Regression (RLR), penalizes the sum of the squared regression coefficients in the optimization goal of the logistic model.
2. Support Vector Machines (SVMs), represent observations as points in space and finds the hyperplane that separates them in the best possible way. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class, since in general the larger the margin the lower the generalization error of the classifier [11]. SVMs can efficiently perform a non-linear classification using the kernel trick, implicitly mapping the inputs into high-dimensional feature spaces. In these experiments, standard parameters are used: radial Kernel; $cost = 1$; $\gamma = 1/p = 0.00625$; $\epsilon = 0.1$.
3. Random Forest [56], is an ensemble learning method that operates by constructing a multitude of decision trees at training time and outputting the mean prediction (in case of regression) of the individual trees. Decision trees partition the factor space according to value tests, therefore resulting in a non-linear classification. In these experiments, standard parameters are used: number of trees = 500; variables tried at each split = $\lfloor p/3 \rfloor$ = 53, where p is the number of features selected in Step 3.
4. Naive Bayes, learns the class-conditional probabilities $p(x_i|y)$ of each input $x_i, i = 1, \ldots, n$ given the class label y. Classification is made by using Bayes' rule to compute the posterior probability of each class y given the vector of observed attribute values. Naive Bayes assumes the features are conditionally independent given the category. Despite its simplicity and the fact that its conditional independence assumption does not hold in real-world situations, Naive Bayes-based Regression still tends to perform surprisingly well [13]. From the Naive Bayes model the output obtained is the probability for the false class.

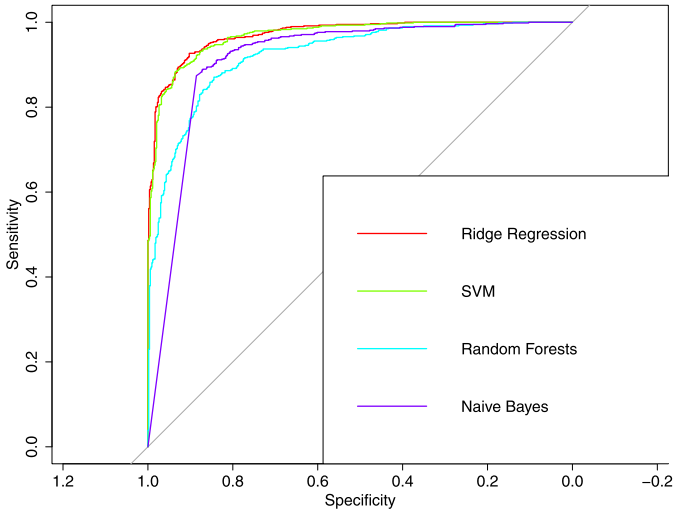

Fig. 2. ROC curves for the models considered. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The performance of the methods, combined with the feature selection procedure described, has been evaluated using using Leave-One-Out Cross-Validation (LOOCV). Table 1 shows some statistics - Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC - calculated according to the following definitions [57]:

- *Accuracy*, $TP + TN/TP + TN + FP + FN$
- *Sensitivity*, $TP/TP + FN$
- *Specificity*, $TN/TN + FP$
- *Precision*, $TP/TP + FP$
- *F1*, $2 * TP/2 * TP + FP + FN$
- *AUC*, Area under the ROC curve, where the ROC curve is created by plotting the true positive rate against the false positive rate at various threshold settings.

The classification into "false" (corresponding to value 1) or "true" (corresponding to value 0) is obtained by rounding the probabilities estimated by the regression methods (i.e., threshold equal to 0.5). Therefore:

- TP (True Positive) is the number of false documents correctly classified as false.
- TN (True Negative) is the number of true documents correctly classified as true.
- FN (False Negative) is the number of false documents wrongly classified as true.
- FP (False Positive) is the number of true documents wrongly classified as false.

Fig. 2 shows the ROC for the considered models. By observing the table and the figure it can be concluded that, overall, RLR

Table 1
Performance statistics for the models considered: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.9153 (0.8975, 0.931)	0.9269	0.9026	0.9129	0.9198	0.9721
SVM	0.9029 (0.884, 0.9196)	0.9065	0.8989	0.9080	0.9072	0.9694
Random Forests	0.852 (0.8299, 0.8723)	0.8810	0.8202	0.8436	0.8619	0.9244
Naive Bayes	0.8253 (0.8018, 0.8471)	0.9660	0.6704	0.7634	0.8529	0.9135

Table 2
Performance statistics of our models on the Positive Sentiment Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.975 (0.9617, 0.9847)	0.9800	0.9700	0.9703	0.9751	0.9955
SVM	0.9688 (0.9542, 0.9797)	0.9750	0.9625	0.9630	0.9689	0.9915
Random Forests	0.8475 (0.8207, 0.8717)	0.8600	0.8350	0.8390	0.8494	0.9326
Naive Bayes	0.8925 (0.8689, 0.9131)	0.8775	0.9075	0.9046	0.8909	0.9414
[12]	0.898	–	–	–	–	–
[37]	0.893	–	–	–	–	–
[9]	–	–	–	–	0.89	–
[39]	–	–	–	–	0.902	–

has the best performance and Naive Bayes has the best Sensitivity and Recall. Also, no significant differences are detected between RLR and SVM in terms of Accuracy (i.e., the confidence interval overlap). In terms of ROC, RLR and SVM clearly dominate the other models, while no clear dominance can be established between them. Multiple DeLong's test for two correlated ROC curves [58] show that no significant differences are detected between the AUC of RLR and SVM ($Z = 1.3206$, p-value $= 0.1866$), while RLR is clearly superior to Naive Bayes ($Z = 7.8948$, p-value $= 2.908e{-}15$) and Random Forests ($Z = 8.2953$, p-value $< 2.2e{-}16$) in terms of AUC.

In conclusion, all the models perform well, however two groups can be identified: RLR and SVM showing high performances, and Random Forests and Naive Bayes presenting slightly worse results. However, RLR has an advantage that sets it apart from SVM, which is that the resulting estimated model can be interpreted easily and used to obtain useful insights, as shown in Section 5.

4.1. Comparison with state-of-the-art methods

The methodology used in *VeriPol* has been applied to several datasets from the literature where the performance of the presented model has been compared against the results of state-of-the art approaches. Concretely this paper's methodology has been tested on the following datasets:

Positive Sentiment Hotel Opinions This dataset, first introduced in [12] , and used in [12,37,39] and [9] , is comprised of 400 deceptive and 400 truthful positive reviews on 20 hotels in the Chicago area. *VeriPol*'s results against the aforementioned state-of-the art approaches are shown in Table 2.

Positive and Negative Sentiment Hotel Opinions This dataset is first introduced in [37] and extends the Positive Sentiment Hotel Opinions dataset, by introducing 400 deceptive and 400 truthful negative reviews on 20 hotels in the Chicago area. Some authors consider the positive and negative dataset separately [9,37,39] (see Table 3), while others jointly [9,37] (see Table 4).

Abortion Opinions This dataset, introduced and used in [33], presents 100 true and 100 deceitful opinions on the topic of abortion. *VeriPol*'s results against its original methodology are shown in Table 5.

Death Penalty Opinions This dataset, introduced and used in [33], presents 100 true and 100 deceitful opinions on the topic of death penalty. *VeriPol*'s results against its original methodology are shown in Table 6.

Best Friend Opinions This dataset, introduced and used in [33], presents 100 true and 100 deceitful opinions on the topic of death penalty. *VeriPol*'s results against its original methodology are shown in Table 7.

Extended Hotel, Restaurant, and Doctor Opinions This dataset, first introduced in [38] , presents 940 true and 940 deceitful opinions on hotels (extending the Positive and negative Sentiment Hotel Opinions dataset), 520 true and 200 deceitful opinions on restaurants, and 400 true opinions on doctors, obtained from different sources (i.e., Turkers, experts and customers). The three domains are considered separately in [38] (see Tables 8–10), and jointly in [14] (see Table 11).

The bottom rows of Tables 2–11 illustrate the performance achieved by methodologies from the literature. Note that only the best result presented in each paper is reported. It can be easily seen that this paper's presented methodology (top rows), specially the Ridge Logistic Regression, outperforms all of them. The reader is reminded that these results were obtained using LOOCV and that *VeriPol* includes filters to remove variables that could introduce overfitting and biases as introduced in Section 3.2. The improvement of more than 7% in all tested datasets in different domains against different state-of-the-art methodologies shows the robustness and value of *VeriPol*.

4.2. Human experts validation

To test the efficiency of *VeriPol* against the manual assessment of falsehood of each registered report, two officers from the Spanish National Police Corps analyzed the reports in the corpus. These two experts where given a random subsample of the corpus comprised of 659 reports and were asked to classify each report as true or false, and evaluate how certain they were of their answer using a 5 star-Likert scale. Note that to avoid biases in the experiment, reports were randomly shuffled, anonymized and untagged before handing them to the agents.

A comparison of the accuracy of the evaluators versus *VeriPol* is given in Table 12. The AUC could not be calculated as the human experts classified the reports as true or false rather than assigning a falsehood probability. For this experiment, the performance of *VeriPol* is computed using LOOCV on the same subsample. In terms of accuracy, the human experts have statistically sim-

Table 3
Performance statistics of our models on the Negative Sentiment Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.95 (0.94, 0.97)	0.95	0.96	0.96	0.95	0.99
SVM	0.95 (0.94, 0.97)	0.95	0.95	0.95	0.95	0.99
Random Forests	0.82 (0.80, 0.85)	0.84	0.80	0.81	0.83	0.91
Naive Bayes	0.87 (0.84, 0.89)	0.85	0.88	0.88	0.87	0.91
[37]	0.86	–	–	–	–	–
[9]	–	–	–	–	0.865	–
[39]	–	–	–	–	0.872	–

Table 4
Performance statistics of our models on the Positive and Negative Sentiment Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.95 (0.93, 0.96)	0.95	0.95	0.95	0.95	0.99
SVM	0.95 (0.93, 0.96)	0.94	0.95	0.95	0.95	0.98
Random Forests	0.85 (0.83, 0.88)	0.86	0.85	0.85	0.86	0.93
Naive Bayes	0.88 (0.85, 0.90)	0.85	0.90	0.90	0.87	0.94
[37]	0.872	–	–	–	–	–
[9]	–	–	–	–	0.879	–

Table 5
Performance statistics of our models on the Abortion Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.80 (0.73, 0.85)	0.87	0.72	0.76	0.81	0.85
SVM	0.76 (0.69, 0.81)	0.82	0.69	0.73	0.77	0.84
Random Forests	0.73 (0.66, 0.79)	0.73	0.73	0.73	0.73	0.79
Naive Bayes	0.68 (0.61, 0.74)	0.93	0.42	0.62	0.74	0.83
[33]	0.70	–	–	–	–	–

Table 6
Performance statistics of our models on the Death Penalty Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.73 (0.67, 0.80)	0.81	0.66	0.71	0.75	0.61
SVM	0.73 (0.67, 0.80)	0.81	0.66	0.71	0.75	0.81
Random Forests	0.73 (0.66, 0.79)	0.81	0.65	0.70	0.75	0.63
Naive Bayes	0.61 (0.54, 0.68)	0.92	0.30	0.57	0.70	0.62
[33]	0.674	–	–	–	–	–

Table 7
Performance statistics of our models on the Best Friend Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.90 (0.85, 0.94)	0.90	0.91	0.91	0.90	0.97
SVM	0.90 (0.85, 0.94)	0.90	0.91	0.91	0.90	0.95
Random Forests	0.78 (0.72, 0.84)	0.74	0.82	0.80	0.77	0.86
Naive Bayes	0.82 (0.76, 0.87)	0.69	0.95	0.93	0.80	0.92
[33]	0.77	–	–	–	–	–

ilar rates (their 95% CI overlap), while *VeriPol* significantly outperforms the human experts (the 95% CI of VeriPol do not overlap with those of the experts) obtaining an improvement of more than 15% versus Evaluator 1 and of more than 20% versus Evaluator 2. Interestingly, Evaluator 2 achieves a slightly better Sensitivity and Recall, however this can be obtained by classifying as "false" most of the reports. Regarding the level of certainty, Fig. 3 shows the number of reports obtaining a certain rating for each evaluate. From the figure it can be seen that Evaluator 1 is more confident in his/her answers and, indeed, achieves a better precision.

According to this analysis, *VeriPol* is capable of detecting hidden patterns that are not evident to the "human eye," even to that of an expert.

4.3. False robbery reports detection model

Table 13 presents, for each variable, the coefficients of the false report detection model (translated from Spanish). For a list of the variables included, please refer to Section 3.2. Variables preceded by a hash character (#) correspond to PoS tags (see Table A.16 for the full list of PoS tags and their description), those preceded by a star character (*) are document statistics (e.g., number of tokens, lemmata, and sentences within a document), and the rest are unigram lemmata. To simplify the presentation, variables are grouped according to the effect on the classification and to the corresponding type (i.e., binary, frequency, logarithm, and ratio, see Section 3.2), as variables of different types are not compara-

Table 8
Performance statistics of our models on the Extended Hotel Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.93 (0.92, 0.94)	0.94	0.92	0.93	0.93	0.98
SVM	0.92 (0.89, 0.95)	0.95	0.90	0.90	0.93	0.98
Random Forests	0.82 (0.80, 0.84)	0.83	0.80	0.81	0.82	0.90
Naive Bayes	0.85 (0.83, 0.87)	0.90	0.80	0.82	0.86	0.91
[38]	0.664	–	–	–	–	–

Table 9
Performance statistics of our models on the Restaurant Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.94 (0.91, 0.96)	0.96	0.92	0.92	0.94	0.98
SVM	0.92 (0.89, 0.95)	0.95	0.90	0.90	0.93	0.97
Random Forests	0.82 (0.78, 0.86)	0.83	0.81	0.82	0.82	0.90
Naive Bayes	0.86 (0.82, 0.89)	0.79	0.93	0.92	0.85	0.91
[38]	0.765	–	–	–	–	–

Table 10
Performance statistics of our models on the Doctor Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.98 (0.97, 0.99)	0.96	1.00	1.00	0.98	1.00
SVM	0.99 (0.98, 1.00)	0.98	1.00	1.00	0.99	1.00
Random Forests	0.94 (0.92, 0.95)	0.87	1.00	1.00	0.93	1.00
Naive Bayes	0.87 (0.84, 0.89)	0.87	0.86	0.85	0.86	0.91
[38]	0.647	–	–	–	–	–

Table 11
Performance statistics of our models on the Extended Hotel, Restaurant, and Doctor Opinions dataset: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1, AUC.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1	AUC
RLR	0.95 (0.94, 0.96)	0.95	0.96	0.96	0.95	0.99
SVM	0.95 (0.94, 0.96)	0.94	0.96	0.96	0.95	0.99
Random Forests	0.84 (0.82, 0.85)	0.84	0.83	0.84	0.84	0.91
Naive Bayes	0.87 (0.85, 0.88)	0.90	0.84	0.85	0.87	0.93
[14]	–	–	–	–	–	0.907

Table 12
Performance comparison between the expert evaluators and *VeriPol* on a random subsample of 659 reports: Accuracy (95% Confidence Interval), Sensitivity, Specificity, Precision, Recall, F1.

	Accuracy (95% CI)	Sensitivity	Specificity	Precision	F1
Evaluator 1	0.7596 (0.7248, 0.792)	0.8556	0.6291	0.7583	0.8040
Evaluator 2	0.7255 (0.6895, 0.7594)	0.9683	0.3883	0.6873	0.8039
VeriPol	0.9272 (0.9046, 0.9458)	0.9556	0.8877	0.9219	0.9385

ble. This is due to the differences in magnitude: binary variables take values 0 or 1; frequency variables can take any positive natural number; logarithm variables are real values on a logarithmic scale; ratio variables are real values between 0 and 1 and, in the dataset considered, they typically take very small values.

The coefficients to the variables can be interpreted to understand their importance when it comes to detect false reports. This, in turn, allows to identify differences between real and false reports, patterns of behavior, and shed some light on how people lie to the police. *VeriPol* computes the probability of falsehood of a report, i.e., predicted values close to 1 indicate that the report has a high probability of being false, and viceversa. Therefore positive coefficients correspond to variables that increase the probability of falsehood, while negative coefficients correspond to variables that decrease this probability. Also, the higher the absolute value, the stronger this effect is. For example, the binary variable associated with the unigram “day” has an associated coefficient equal to 0.4825, meaning that the appearance of the word “day” in a document slightly increases its probability of falsehood.

In the following, the variables chosen by *VeriPol*, their corresponding coefficients and their interpretation are given.

The intercept of the RLR model is 0.1513 , therefore initially all the reports are considered presumably true. According to *VeriPol*, reports that present the words “day,” “lawyer,” “to report” and “to extract” are more likely false, while the words “bus,” “to kill,” “to drag” and ”pendant“ characterize true reports. Furthermore, a high appearance of the words “two hundred,” “felony,” “hand bag” and of subordinate conjunctions that introduce finite clauses (corresponding to variable CSUBF, e.g., “barely”) is typical in false reports, whereas a high frequency of the words “Grand” (appearing in the corpus as part of the sentence “Samsung Galaxy Grand”), “plate” (appearing in the corpus with the meaning “vehicle registration plate”), “around,” and “taken” (appearing in the corpus with the meaning “stolen”) indicates that the report is probably true.

Table 13
Model variables and corresponding coefficients. Translated from Spanish. Variables preceded by a hash character (#) correspond to PoS tags (see Table A.16 for the full list of PoS tags and their description), those preceded by a star character (*) are document statistics (e.g., number of tokens, lemmata, and sentences within a document), and the rest are unigram lemmata.

Falsehood variables				Truth variables			
Binary variable	Frequency variable	Logarithm variable	Ratio variable	Binary variable	Frequency variable	Logarithm variable	Ratio variable
Day (0.4825)	Two hundred (0.2996)	To pull out (0.3940)	Responsible (32.6211)	Bus (−0.5166)	Grand (−0.3657)	Officer (−0.3786)	Even (−72.3152)
Lawyer (0.4307)	#CSUBF (0.1910)	Shove (0.2534)	Disregard (29.9065)	To kill (−0.4897)	Plate (−0.1898)	Taken (−0.2527)	To escape (−68.6025)
To report (0.4300)	Felony (0.1677)	Insurance (0.2433)	Separate (28.6956)	To drag (−0.4837)	Around (−0.1864)	To pass (−0.2429)	Chinese (−52.5844)
To extract (0.4155)	Hand bag (0.1275)	Back (0.1442)	#NEG (28.6044)	Pendant (−0.3778)	Taken (−0.1617)	Final (−0.2422)	Doubt (−50.2013)
To start (0.3684)	Cash (0.1166)		Helmet (26.9261)	Chinese (−0.3666)	Light (−0.1506)	Police (−0.1664)	Turning up (−48.1339)
To react (0.3120)	List (0.1129)		Iphone (25.5605)	Landing (−0.3639)	Police (−0.0972)	Again (−0.1522)	Pay attention to (−47.6469)
Policy (0.2389)	Backpack (0.1070)		To report (20.7153)	Assault (−0.3517)	#VCLIger (−0.0824)	#VSinf (−0.1493)	To shout (−40.6734)
Apple (0.2352)			Unique (20.6342)	Beard (−0.3425)	Friend (−0.0612)	Face (−0.1450)	Authorised (−34.6212)
Company (0.2195)			Contract (19.1924)	Report (−0.3413)	Aforementioned (−0.0546)	Short (−0.1434)	To tend (−32.1409)
Only (0.2021)			Cash (19.1827)	Mountain (−0.3215)	Male (−0.0250)	#PAL (−0.1351)	Climb (−31.3857)
Insurance (0.1918)			Shoulder (17.9951)	Officer (−0.3093)	#PPC (−0.0110)	Friend (−0.1327)	Car (−28.1087)
Establishment (0.1853)			Removal (17.6851)	Wrestle (−0.2820)		Age (−0.1157)	Doctor (−16.7533)
To contain (0.1741)			Series (16.8249)	Would recognise (−0.2722)		#VCLIinf (−0.1079)	Neck (−16.6258)
Model (0.1274)			Urban (16.5677)	Entrance hall (−0.2617)			Theft (−16.1871)
Back (0.1198)			Model (14.4221)	Neighbor (−0.2614)			Chain (−16.0589)
Behind (0.1172)			Next (13.5029)	Turning up (−0.2404)			#PPX (−15.1829)
Number (0.1148)			Specifically (13.0455)	To start (−0.2283)			#DM (−7.8174)
Black (0.1035)			To detail (12.5800)	Four (−0.1957)			#ART (−4.1312)
			List (10.0908)	Several (−0.1696)			
			Right (9.4583)	Attach (−0.1628)			
			Back (7.7415)	Record (−0.1627)			
			Mobile phone (7.5446)	Neck (−0.1600)			
			To carry (7.4077)	Make known (−0.1362)			
			Object (7.2053)	To grab (−0.1275)			
			*Document Sentences (6.8603)	Short (−0.1251)			
			Euro (6.8160)	Hair (−0.1043)			
			#NC (1.6161)	Brunette/dark-skinned (−0.1007)			
			*Document Concepts (1.1135)	To recognise (−0.1002)			
				Thin (−0.0945)			
				Centimeter (−0.0941)			
				Constitution (−0.0907)			

Additionally, high magnitudes of the words “to pull out,” “shove,” “insurance,” and “back” (appearing in the corpus as a the body part) indicate that the report is false, while high magnitudes of the words “officer” (appearing in the corpus with the meaning “Police officer”), again “taken,” “to pass,” and “final” indicate that the report is true. Finally, high ratios in the document of the words “responsible,” “disregard” (appearing in the corpus as part of the sentence “disregard of its truthfulness”), “separate,” and negations (variable NEG) are typical in false reports, whereas high ratios of “even,” “to escape,” “Chinese,” and “doubt” (appearing in the corpus as part of the sentence “without any doubt,” thus taking the opposite meaning).

As mentioned, *VeriPol* estimates a probability of falsehood. The histogram of the probabilities predicted by *VeriPol* relative to the reports in the dataset is presented in Fig. 4. It can be appreciated the U-shape indicating that most values are close to either 0 or 1, rather than 0.5, suggesting that *VeriPol* provides a definitive answer most of the time. When using *VeriPol* as a classifier (with a threshold equal to 0.5) the accuracy (computed using LOOCV) is 0.9153 and the corresponding confusion matrix can be observed in Table 14. In the application context considered, the most sen-

Table 14
Confusion matrix of *VeriPol*. Columns correspond to the reference (i.e., the real document label) while the rows correspond to the prediction.

	0	1
0	482	43
1	52	545

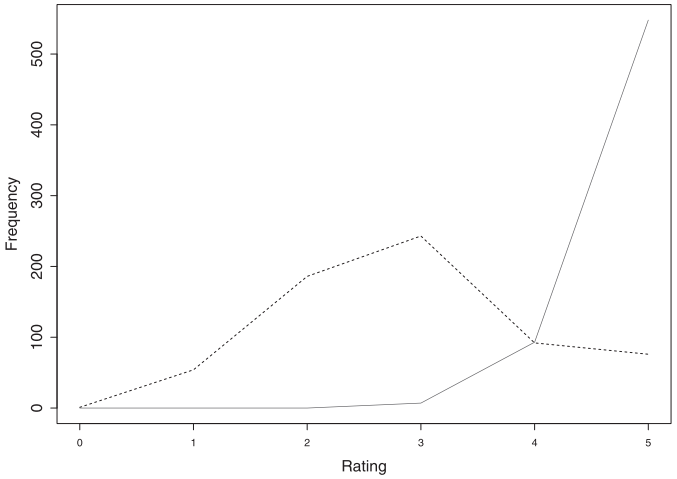

Fig. 3. Evaluators certainty ratings. Evaluator 1 is represented with a continuous line while Evaluator 2 is represented with a dashed line.

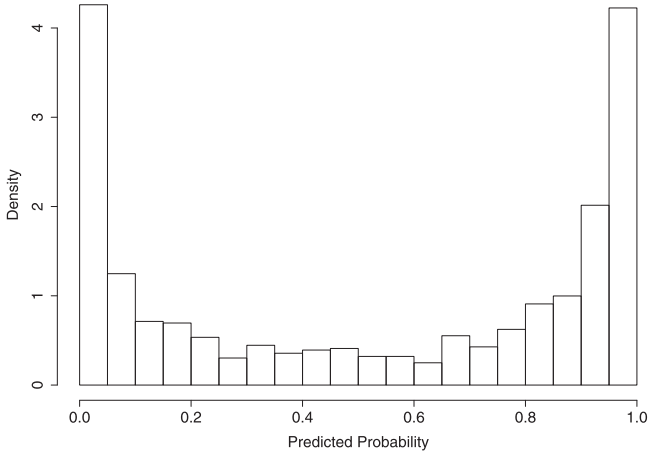

Fig. 4. Histogram of *VeriPol*'s predicted probabilities.

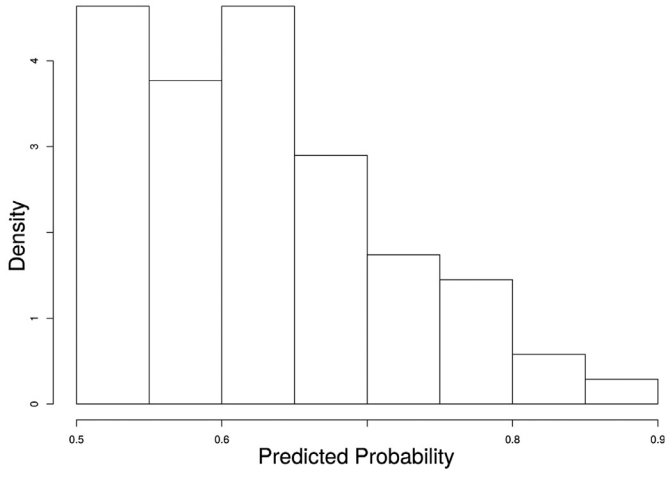

Fig. 5. Histogram of *VeriPol*'s predicted probabilities for false positive documents.

sible type of error is the error of the first kind, that correspond to false positives, i.e., true reports that are classified as false. The ratio of false positive for *VeriPol* is 0.0954. Fig. 5 shows the probability histogram for the false positive cases. It can be clearly seen that the highest percentage of cases have a probability close to 0.5, that is, the answer provided by *VeriPol* is rather uncertain. For this reason, it is recommended that the predicted probability is interpreted by agents that can adjust the classification threshold according to their experience. Also, the predicted probability could be used as a ranking factor among the reports to assign priority to cases, rather than for proper classification, since there exists an obvious trade-off between accuracy and false-positive rate.

5. Discussion: model analysis

From the analysis of *VeriPol*'s features and coefficients it is possible to draw interesting insights on how people lie to the police. In fact, the model is capable of discerning significant differences in the narrative of true and false reports that lead to the best separation between these two classes. Reviewing all the concepts included in *VeriPol* and the situations where they appear in the corpus allows to understand and extract the characteristics identified by the model. From this analysis it surfaced that true and false reports mostly differ in three main aspects: i) *modus operandi* of the aggression, ii) morphosyntax of the report and iii) amount of details. Note that the concrete variables used to guide the following insights are highlighted in the explanation with quotation marks.

5.1. Modus operandi

With respect to the *modus operandi* of the aggression described in the report it can be found that reports that describe the following type of felonies have lower probabilities of falsehood: i) Theft of necklaces (mostly jewelry or gold chains) represented in the reports by: "neck," "chain," or "pendant." ii) Crimes that involve "mountain" bikes, either as the object stolen or as the mean of transportation of the aggressor. iii) Aggressions close to the home of the victim, near the "entrance hall" or in the "landing" of the stairs.

On the other hand, those that describe the following type of felonies have a higher probability of falsehood: i) Pulls or "shoves" from "behind" of the victim's "backpack," "hand bag," or of items hanging from the victim's "shoulder." In general, any aggressions coming from the "back." ii) Attacks by someone that is wearing a motorcycle "helmet." iii) Robberies of expensive "mobile phones," making special stress on the brands "iPhone," "Apple," Samsung Galaxy "Grand." Other concepts identified by *VeriPol* that belong to this group are "model" (of the phone), "series" (of the phone), phone "company," or the type of phone "contract."

5.2. Morphosyntax

When looking at the morphosyntactic characteristics of reports, it is observed that truthful reports tend to have the following peculiarities: i) High frequency of clitic gerund verbs (variable VCLIger), clitic infinitive verbs (variable VCLinf), Clitics and personal pronouns (variable PPX) and Clitic personal pronouns (variable PPC). ii) High frequency of Demonstrative pronouns (variable DM) and Articles (variable ART). iii) Very high occurrences of the verb "to be" in infinitive (variable VSinf) and the portmanteau word, "to"(variable PAL). The high occurrence in truthful reports of both clitic verbs and nouns (and also of the verb "to be") represents a high presence of descriptions of interactions between actors in a narrative. More importantly it describes reflexive actions happening both to the victim or the aggressor. This fact highly contrasts with the description of false reports shown next, where common and not reflexive nouns and verbs are predominant. This means that truthful reports are more centered in the story telling of the dynamic of the felony and the interactions of both the victim and the aggressor where clitic verbs and nouns represent that the action falls on the subject of the sentence. It is a closer and more personal way of narrating the incident. Also, the high frequency of

demonstrative pronouns, articles or the word "to" in comparison with the absence thereof in false reports reflects that truthful reports tend to be richer in descriptions.

On the contrary, false reports tend to present the following syntactic characteristics. i) High frequency of subordinating conjunctions that introduce finite clauses (variable CSUBF, i.e. "barely"). This characteristic reflects that false reports tend to present sentences that reflect lack of information being "barely" most of the times part of sentences in the line of "s/he could barely see" or "s/he barely remembers." Similarly, false reports also present high ratios of negations (variable NEG) in comparison with the length of the document. Examples of sentences found with this characteristic are: "s/he cannot report more data," "s/he has not suffered injuries," "s/he could not see," "s/he could not recognize" or "s/he did not attend a medical facility." Interestingly, as shown next, these very same sentences but in positive are very representative of truthful reports. ii) High percentage of the variable Document Sentences, which is the inverse of the average length of sentences. Meaning that false reports seem to be characterized by shorter sentences. iii) Finally, and contrarily to truthful reports that seem to be delineated by reflexive nouns and verbs indicating that the action is being focused on the actor of the sentences, high ratios of common nouns (variable NC) are identified as properties of false reports. While truthful reports are characterized by high frequency of reflexive verbs and nouns reflecting a personal description of actions happening to the victim, false reports are delineated by enumerations of common nouns, focusing more on objects rather than the dynamic of the felony. This is also confirmed by a higher ratio of Document Concepts in false reports, that is unique concepts, meaning that false reports are more similar to lists of enumerated facts or objects than true reports, that focus more on the description of a limited number of concepts.

5.3. Amount of details

With respect to the amount and type of information provided, truthful reports tend to have the following peculiarities. i) Longer documents that provide more details about the robbery and the aggressor. This conclusion can be drawn from the coefficients associated to variables representing the ratios and frequencies of "even," "(without) doubt" (note in the case of truthful reports doubt is always accompanied by without) and "aforementioned." Also, from the appearance of the words "would recognize" or "to recognize," "pay attention to" (a detail), meaning that the victim is able to recognize the attacker. Finally, from the presence of characteristics describing a person or the attack with words like "face," "beard," "hair," "centimeter," "brunette/dark-skinned," "thin," "constitution," "short," "light," the high presence of "age," "male," or vehicle "plate." ii) Similarly, verbs indicating interaction between the attacker and the victim are signs of truthful reports: "to shout," "to wrestle," "to grab," "to start to." Again, this goes along with the previous conclusion that the more information and details, the lower the probability of falsehood. iii) Finally, it seems that having interacted during or after the incident with other people or being able to justify or provide extra information are also good signs to identify truthful reports. Such as: a) When the victim includes or reports extra information with words like "attach," " to make known" (appearing in the corpus as part of the sentence "wishes to place on record"). b) When the victim informs that s/he received medical care after the felony. The words that represent this situation are "doctor," medical "report," "authorised" (appearing in the corpus indicating a document signed by an authorized/empowered person), "to tend" (appearing in the corpus with the meaning "to be tended"). c) When the victim has reported the presence of witnesses like a "friend" or a "neighbor." d) When there is a high frequency of the words "police," "officer" or "turning up" (referred in the corpus to an officer). This represents that the victim has looked for or contacted security forces right after the aggression.

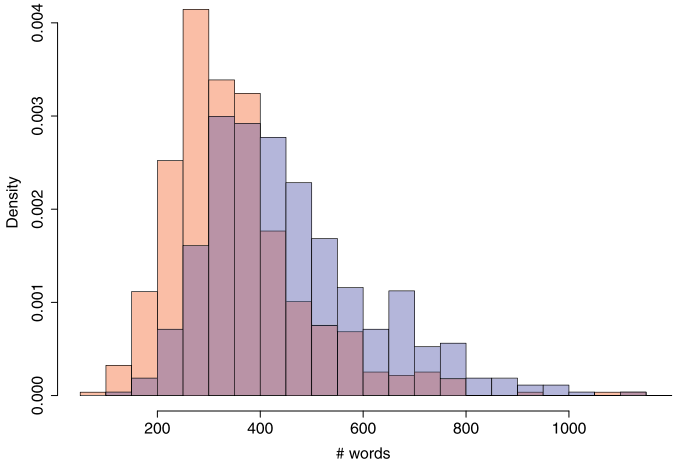

Fig. 6. Histogram of the length of the reports. True reports are represented in blue and false reports are depicted in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

On the other hand, giving vague information about the attack is distinctive of false reports. Some of the words that represent this situation are "back," "behind," "helmet," referring to the situations where the victim could not see the attacker. Also, references to "black" clothes are more likely in false reports. The word "only" denotes vague descriptions as it usually appears in sentences like "only being able to" or "only having seen." Other words suggest a high interest of the victim in claiming money to insurance companies, denoting the intention of the complainant in filing a false robbery report to commit a fraud: "policy," "insurance," "iPhone," "Apple," "Grand," "mobile phone," " model," "series," "euro," "cash," "two hundred," "company," "contract" (with a mobile company). Finally, the ratio of "disregard" is a characteristic of false reports. The word "disregard" appears in all the reports as part of a standard sentence that explains the legal consequences of giving a statement in "disregard of the truth." Therefore, its ratio is a proxy to the inverse length of the text. This means that shorter documents are more likely to be false. This is confirmed by Fig. 6 that shows in blue and red the histograms of the number of words in true and false reports, respectively. The data proves that true reports contain more words and false reports tend to be shorter.

6. Pilot study

As explained, *VeriPol* has been designed in collaboration with the Spanish National Police that provided the report corpus and the resources to run a pilot study.

To test the efficacy and effectiveness of *VeriPol*, a pilot study has been undertaken in the urban areas of Murcia and Málaga, Spain. More in detail, the pilot study was run in Murcia (four police departments involved) from the 5th to the 9th of June 2017, while it took place in Málaga (six police departments involved) from the 12th to the 16th of June 2017. In each destination, two expert officers in false report detection and in *VeriPol* were sent to install the software, give a short course on its use to the local agents and investigators, and supervise all the activity. After that, all the new robbery reports and all the open robbery cases of 2017 were evaluated by *VeriPol*.

The results of the pilot study are shown in Table 15. As it can be observed, the implementation of *VeriPol* allowed for an impressive increase in productivity in terms of number of false cases of robbery detected and successfully closed. In fact, in just one week, 25

Table 15
Results of the pilot study: urban area of destination, number of false violent robbery cases closed during the pilot study, average number of false violent robbery cases closed in the month of June 2008–2016, and ratio of false violent robbery cases closed and suspects interrogated after *VeriPol* analyzed the report and assigned it a high probability of falsehood.

Destination	# cases closed pilot study	Avg # cases closed June	Ratio(%)
Murcia	31	3.33	81.58
Málaga	49	12.14	84.78
Total	80	15.47	83.54

and 39 false robbery reports were detected and closed while the average number of false robbery cases detected and closed in the months of June 2008–2016 is 3.33 and 12.14, in Murcia and Málaga respectively. More importantly, the success ratios (i.e., the number of false robbery cases closed divided by the number of complainants interrogated) are 81.58% in Murcia and 84.78% in Málaga. It is important to notice that a robbery case is successfully closed as false only if the complainant confesses his/her crime. Therefore, these ratios are necessarily lower than or equal to the real precision of *VeriPol* during the pilot study, as potentially, not all the guilty complainants might have confessed. As a consequence, the results of the pilot study allow to estimate a lower bound on the real precision. In particular, *VeriPol* correctly evaluated at least 83.54% of the reports classified as false.

Also, to understand the level of acceptance associated with the use of *VeriPol*, all the agents and officials that participated in the pilot study were asked to answer to an anonymous questionnaire on a voluntary basis that addressed their level of satisfaction with *VeriPol* and the perceived usefulness. Overall, the 21 participants agreed that *VeriPol* is useful, easy to use, and that it should be extended to other types of crime.

Given the successful outcome of the pilot study, *VeriPol* is currently being installed in the report-managing software of the SNP as a decision support tool. It is important to remark that all the decisions regarding how to proceed with the investigation based on the probability estimated by *VeriPol* are left to the officers and are their responsibility.

7. Conclusions and future research

In this paper a text-based lie detection model for police reports is presented. The model consists of a multi-step methodology that combines NLP and ML techniques including among others feature selection by L1 penalization and heuristic rules. Our findings, obtained by analyzing the structure of the designed model, *VeriPol*, are reliable as it presents a very low misclassification error, lower than 9%, while expert policemen showed an error of 25%, approx. *Also, the model improves the accuracy of several models from the literature in various domains.* A pilot study allowed to verify the performance of *VeriPol* on the field, proving its usefulness by allowing to solve a very high number of false robbery cases. In particular, a lower bound on the empirical precision of *VeriPol* during the pilot study was 83%, approx.

VeriPol's analysis makes possible to identify a clear pattern. Truthful reports usually present more details, personal information and descriptions. On the other hand, false reports are mostly characterized by being shorter and more focused on the stolen property rather than the aggression and by the impossibility of providing precise information about the incident, recognize the attacker, producing witnesses, or other hard evidence (e.g., contacting a police officer right after the aggression or a doctor).

The findings provided by *VeriPol* coincide with those in other researches in the domain of fake online reviews. Ott et al. [12] reported that truthful opinions tend to include more sensorial and concrete language than deceptive opinions; in particular, truthful opinions are more specific about spatial configurations. Vrij et al. [40] concluded that liars have considerable difficultly encoding spatial information into their lies which seems to relate to our conclusion that truthful reports include more details and descriptions than false reports. Regarding Newman et al. [3] analysis about categories and type of words in false stories, it is impossible to compare our linguistic analysis to theirs (count of present tense, past tense, pronouns, etc) for two main reasons: i) police reports are written by an officer reflecting the description of the event of a victim while the study by Newman et al. concerns self written deceptive texts and ii) the presence of adjectives, nouns, pronouns, etcetera, depends on the topic. However, these findings suggest that, regardless of the domain, untruthful texts could be identified by lower cognitive complexity, fewer self-references, more words and more negative words. Future research on this topic should investigate the degree of veracity of this statement.

Looking at the practical implications of this research, to the Spanish National Police *VeriPol* represents an advancement in terms of methodology, as well as a complete change of paradigm in the way that investigations are carried out. More specifically, the practical contribution of *VeriPol* is three-fold: i) The existence of this decision support system, if properly advertised, can help in discouraging citizens from filing a false report, hence preventing a crime; ii) it improves the use of limited police resources by supporting investigations; iii) it results in a reduction of the noise due to false reports in police databases by helping clearing false robbery cases.

In addition, this system opens the door to a new form of investigation that is individualized in the robbery crimes due its special incidence and social repercussion and with the possibility of extending it to any type of police investigation as we believe that this is the beginning of a unusual and innovative model.

Finally, the implementation of *VeriPol* makes the Spanish National Police the first in the world to use a decision support system of this type. In fact, there is no other system with similar characteristics, neither at academic nor at practitioner level, and research on detecting lies from text is still in its infancy. Similar initiatives in foreign police have had a very strong media impact. For example, the PredPol tool developed between the California Santa Cruz Police and the University of California Los Angeles has been named by Time Magazine as one of the 50 Best Inventions of 2011 [59]. The implementation of the model proposed in this investigation puts the National Police in the forefront as one of the most advanced police in the world, with a very positive impact for its image, both nationally and internationally.

As future work we intend to extend the model and train it in different types of reports to expand its applicability. There is of course much to be done in crime forecasting. However, we hope that this work will be a useful source of ideas for future research and other police agencies, helping reduce crime and increase police effectiveness[3].

Acknowledgments

The research of Liberatore was supported by MINECO (Grant no. MTM2015-65803-R). Camacho-Collados' research is supported by a Google Doctoral Fellowship on Natural Language Processing, 2016. All financial supports are gratefully acknowledged. The information and views set out in this paper are those of the author(s) and do not necessarily reflect the official opinion of the financial support-

[3] All data needed to evaluate the conclusions in the paper are present in the paper. Additional data related to this paper may be requested from the authors. The data for this study is publicly available from http://portal.uc3m.es/portal/page/portal/ifibid/people/quijano/lie_detect.

ers. We would like to thank the Spanish National Police Corps and, in particular, Officer Romera-Juarez for his participation in all the phases of the project and Commissioner Álvarez for his support in the initial stages and for believing in *VeriPol* since the beginning. A special thanks to Commissioners Florentino Villabona and José Antonio Mateos for promoting *VeriPol* in the *Ministerio del Interior* (Spanish Ministry of Interior) and providing all the resources necessary to the development of the pilot study, as well as supporting the implementation of *VeriPol* in the Spanish National Police. Also, the authors would like to thank Agents Francisco Sánchez Merelo and Luis Miguel Martínez Gómez for participating as evaluators. Finally, the authors would like to thank the "Fundación Policía Española" for giving *VeriPol* the "National Police Research Award" 2016-17.

Appendix A

Table A.16
PoS tags and their description (Source: http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/spanish-tagset.txt).

Tag	Description	Tag	Description
ACRNM	Acronym (ISO, CEI)	PPX	Clitics and personal pronouns (nos, me, nosotras, te, sí)
ADJ	Adjectives (mayores, mayor)	PREP	Negative preposition (sin)
ADV	Adverbs (muy, demasiado, cómo)	PREP	Preposition
ALFP	Plural letter of the alphabet (As/Aes, bes)	PREP/DEL	Complex preposition "después del"
ALFS	Singular letter of the alphabet (A, b)	QU	Quantifiers (sendas, cada)
ART	Articles (un, las, la, unas)	REL	Relative pronouns (cuyas, cuyo)
CARD	Cardinals	SE	Se (as particle)
CC	Coordinating conjunction (y, o)	SYM	Symbols
CCAD	Adversative coordinating conjunction (pero)	UMMX	Measure unit (MHz, km, mA)
CCNEG	Negative coordinating conjunction (ni)	VCLIger	Clitic gerund verb
CODE	Alphanumeric code	VCLIinf	Clitic infinitive verb
CQUE	Que (as conjunction)	VCLIfin	Clitic finite verb
CSUBF	Subordinating conjunction that introduces finite clauses (apenas)	VEadj	Verb estar. Past participle
CSUBI	Subordinating conjunction that introduces infinite clauses (al)	VEfin	Verb estar. Finite
CSUBX	Subordinating conjunction underspecified for subord-type (aunque)	VEger	Verb estar. Gerund
DM	Demonstrative pronouns (ésas, ése, esta)	VEinf	Verb estar. Infinitive
FO	Formula	VHadj	Verb haber. Past participle
FS	Full stop punctuation marks	VHfin	Verb haber. Finite
INT	Interrogative pronouns (quiénes, cuántas, cuánto)	VHger	Verb haber. Gerund
ITJN	Interjection (oh, ja)	VHinf	Verb haber. Infinitive
NC	Common nouns (mesas, mesa, libro, ordenador)	VLadj	Lexical verb. Past participle
NEG	Negation	VLfin	Lexical verb. Finite
NMEA	Measure noun (metros, litros)	VLger	Lexical verb. Gerund
NMON	Month name	VLinf	Lexical verb. Infinitive
NP	Proper nouns	VMadj	Modal verb. Past participle
ORD	Ordinals (primer, primeras, primera)	VMfin	Modal verb. Finite
PAL	Portmanteau word formed by a and el	VMger	Modal verb. Gerund
PDEL	Portmanteau word formed by de and el	VMinf	Modal verb. Infinitive
PE	Foreign word	VSadj	Verb ser. Past participle
PNC	Unclassified word	VSfin	Verb ser. Finite
PPC	Clitic personal pronoun (le, les)	VSger	Verb ser. Gerund
PPO	Possessive pronouns (mi, su, sus)	VSinf	Verb ser. Infinitive

References

[1] J. Hirschberg, C.D. Manning, Advances in natural language processing, Science 349 (6245) (2015) 261–266.
[2] M.I. Jordan, T.M. Mitchell, Machine learning: trends, perspectives, and prospects, Science 349 (6245) (2015) 255–260.
[3] M.L. Newman, J.W. Pennebaker, D.S. Berry, J.M. Richards, Lying words: predicting deception from linguistic styles, Personal. Soc. Psychol. Bull. 29 (5) (2003) 665–675.
[4] J.T. Hancock, L.E. Curry, S. Goorha, M. Woodworth, On lying and being lied to: a linguistic analysis of deception in computer-mediated communication, Discourse Process. 45 (1) (2007) 1–23.
[5] J. Cohen, W.L. Gorr, A.M. Olligschlaeger, Leading indicators and spatial interactions: a crime-forecasting model for proactive police deployment, Geogr. Anal. 39 (1) (2007) 105–127.
[6] C.-H. Yu, M.W. Ward, M. Morabito, W. Ding, Crime forecasting using data mining techniques, in: Proceedings of the International Conference on Data Mining, ICDM, IEEE, 2011, pp. 779–786.
[7] E. Fitzpatrick, J. Bachenko, T. Fornaciari, Automatic detection of verbal deception, Synth. Lect. Human Lang. Technol. 8 (3) (2015) 1–119.
[8] Automatic identification and verification of claims in political debates, CLEF 2018 Conference and Labs of the Evaluation Forum. http://alt.qcri.org/clef2018-factcheck/.
[9] L.C. Cagnina, P. Rosso, Detecting deceptive opinions: intra and cross-domain classification using an efficient representation, Int. J. Uncertain. Fuzziness Knowl. Based Syst. 25 (Suppl. 2) (2017) 151–174.
[10] B. Heredia, T.M. Khoshgoftaar, J.D. Prusa, M. Crawford, Cross-domain sentiment analysis: An empirical investigation, in: Proceedings of the IEEE International Conference on Information Reuse and Integration, IRI, 2016, pp. 160–165.
[11] L. Zhou, Y. Shi, D. Zhang, A statistical language modeling approach to online deception detection, IEEE Trans. Knowl. Data Eng. 20 (8) (2008) 1077–1081.
[12] M. Ott, Y. Choi, C. Cardie, J.T. Hancock, Finding deceptive opinion spam by any stretch of the imagination, in: Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, Association for Computational Linguistics, 2011, pp. 309–319.
[13] F. Li, M. Huang, Y. Yang, X. Zhu, Learning to identify review spam, in: Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI, 22, 2011, pp. 2488–2493.
[14] B. Heredia, T.M. Khoshgoftaar, J.D. Prusa, M. Crawford, An investigation of ensemble techniques for detection of spam reviews, in: Proceedings of the IEEE International Conference on Machine Learning and Applications, ICMLA, 2016, pp. 127–133.
[15] P. Rosso, L.C. Cagnina, Deception detection and opinion spam, in: A Practical Guide to Sentiment Analysis, Springer, 2017, pp. 155–171.
[16] M. Crawford, T.M. Khoshgoftaar, J.D. Prusa, A.N. Richter, H. Al Najada, Survey of review spam detection using machine learning techniques, J. Big Data 2 (1) (2015) 23.
[17] A. Heydari, M. ali Tavakoli, N. Salim, Z. Heydari, Detection of review spam: a survey, Expert. Syst. Appl. 42 (7) (2015) 3634–3642.
[18] G.H. Gudjonsson, J.F. Sigurdsson, B.B. Asgeirsdottir, I.D. Sigfusdottir, Custodial interrogation: what are the background factors associated with claims of false confession to police? J. Forensic Psychiatr. Psychol. 18 (2) (2007) 266–275.
[19] D. Lisak, L. Gardinier, S.C. Nicksa, A.M. Cote, False allegations of sexual assault: an analysis of ten years of reported cases, Viol. Against Women 16 (12) (2010) 1318–1334.
[20] R.J. Ofshe, R.A. Leo, The social psychology of police interrogation: the theory and classification of true and false confessions, Stud. Law Polit. Soc. 16 (1997) 189–254.
[21] R.J. Kane, Patterns of arrest in domestic violence encounters: identifying a police decision-making model, J. Crim. Justice 27 (1) (1999) 65–79.
[22] W.L. Perry, Predictive policing: The role of crime forecasting in law enforcement operations, Rand Corporation, 2013.
[23] W. Gorr, R. Harries, Introduction to crime forecasting, Int. J. Forecast. 19 (4) (2003) 551–555.
[24] M. Hvistendahl, Crime forecasters, Science 353 (6307) (2016) 1484–1487, doi:10.1126/science.353.6307.1484.
[25] M. Camacho-Collados, F. Liberatore, A decision support system for predictive police patrolling, Decis. Support Syst. 75 (2015) 25–37.
[26] F. Liberatore, M. Camacho-Collados, A comparison of local search methods for the multicriteria police districting problem on graph, Math. Probl. Eng. 2016 (2016).
[27] B. Baesens, T.V. Gestel, S. Viaene, M. Stepanova, J. Suykens, J. Vanthienen, Benchmarking state-of-the-art classification algorithms for credit scoring, J. Oper. Res. Soc. 54 (6) (2003) 627–635.
[28] G. Miller, Fmri lie detection fails a legal test, Science 328 (5984) (2010) 1336–1337.
[29] C.L. Toma, J.T. Hancock, Reading between the lines: Linguistic cues to deception in online dating profiles, in: Proceedings of the International Conference on Computer Supported Cooperative Work, CSCW, ACM, 2010, pp. 5–8, doi:10.1145/1718918.1718921.
[30] B.M. DePaulo, J.J. Lindsay, B.E. Malone, L. Muhlenbruck, K. Charlton, H. Cooper, Cues to deception, Psychol Bull 129 (1) (2003) 74–118.
[31] V.L. Rubin, Y. Chen, N.J. Conroy, Deception detection for news: three types of fakes, Proc. Assoc. Inf. Sci. Technol. 52 (1) (2015) 1–4.
[32] L. Zhou, J.K. Burgoon, J.F. Nunamaker, D. Twitchell, Automating linguistics-based cues for detecting deception in text-based asynchronous computer-mediated communications, Group Decis. Negotiation 13 (1) (2004) 81–106.
[33] R. Mihalcea, C. Strapparava, The lie detector: explorations in the automatic recognition of deceptive language, in: Proceedings of the International Joint Conference on Natural Language Processing, AFNLP, 2009, pp. 309–312.
[34] T. Fornaciari, M. Poesio, Automatic deception detection in italian court cases, Artif. Intell. law 21 (3) (2013) 303–340.
[35] D. Hernández Fusilier, M. Montes-y Gómez, P. Rosso, R. Guzmán Cabrera, Detecting positive and negative deceptive opinions using PU-learning, Inf. Process. Manag. 51 (4) (2015) 433–443.
[36] J.W. Pennebaker, C.K. Chung, M. Ireland, A. Gonzales, R.J. Booth, The Development and Psychometric Properties of LIWC2007, 2007.
[37] M. Ott, C. Cardie, J.T. Hancock, Negative deceptive opinion spam., in: Proceedings of the HLT-NAACL, 2013, pp. 497–501.
[38] J. Li, M. Ott, C. Cardie, E. Hovy, Towards a general rule for identifying deceptive opinion spam, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 1, 2014, pp. 1566–1576.
[39] D. Hernández Fusilier, M. Montes-y Gómez, P. Rosso, R. Guzmán Cabrera, Detection of opinion spam with character n-grams, in: Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics, Springer, 2015, pp. 285–294.
[40] A. Vrij, S. Leal, P.A. Granhag, S. Mann, R.P. Fisher, J. Hillman, K. Sperry, Outsmarting the liars: the benefit of asking unanticipated questions, Law Hum. Behav. 33 (2) (2009) 159–166.
[41] D. Jurafsky, J.H. Martin, Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, Prentice Hall PTR, 2000.
[42] H. Schmid, Improvements in part-of-speech tagging with an application to german, in: Proceedings of the ACL SIGDAT-workshop, Citeseer, 1995.
[43] T. Mullen, N. Collier, Sentiment analysis using support vector machines with diverse information sources, in: Proceedings of the EMNLP, 4, 2004, pp. 412–418.
[44] M. Toman, R. Tesar, K. Jezek, Influence of word normalization on text classification, Proceedings of InSciT 4 (2006) 354–358.
[45] S. Hassan, R. Mihalcea, C. Banea, Random walk term weighting for improved text classification, Int. J. Semant. Comput. 1 (04) (2007) 421–439.
[46] A.K. Uysal, S. Gunal, The impact of preprocessing on text classification, Inf. Process. Manag. 50 (1) (2014) 104–112.
[47] T. Bolukbasi, K.-W. Chang, J.Y. Zou, V. Saligrama, A.T. Kalai, Man is to computer programmer as woman is to homemaker? debiasing word embeddings, in: Proceedings of the Advances in Neural Information Processing Systems, 2016, pp. 4349–4357.
[48] R. Rudinger, C. May, B. Van Durme, Social bias in elicited natural language inferences, in: Proceedings of the First ACL Workshop on Ethics in Natural Language Processing, 2017, pp. 74–79.
[49] J. Zhao, T. Wang, M. Yatskar, V. Ordonez, K.-W. Chang, Men also like shopping: reducing gender bias amplification using corpus-level constraints, in: Proceedings of the Empirical Methods in Natural Language Processing, 2017.
[50] A. Mukherjee, V. Venkataraman, B. Liu, N.S. Glance, What yelp fake review filter might be doing? in: Proceedings of the International Conference on Weblogs and Social Media, ICWSM, 2013.
[51] L. Gao, S. Zhou, J. Guan, Effectively classifying short texts by structured sparse representation with dictionary filtering, Inf. Sci. 323 (2015) 130–142.
[52] R. Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc. Ser. B Methodol. 58 (1) (1996) 267–288.
[53] J. Friedman, T. Hastie, R. Tibshirani, Regularization paths for generalized linear models via coordinate descent, J. Stat. Softw. 33 (1) (2010) 1.
[54] A.E. Hoerl, R.W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, Technometrics 12 (1) (1970) 55–67.
[55] M.S. Hajmohammadi, R. Ibrahim, A. Selamat, H. Fujita, Combination of active learning and self-training for cross-lingual sentiment classification with density analysis of unlabelled samples, Inf. Sci. 317 (2015) 67–77.
[56] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[57] D.M.W. Powers, Evaluation: from precision, recall and f-measure to roc., informedness, markedness & correlation, J. Mach. Learn. Technol. 2 (1) (2011) 37–63.
[58] E.R. DeLong, D.M. DeLong, D.L. Clarke-Pearson, Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach, Biometrics (1988) 837–845.
[59] L. Grossman, C. Brock-Abraham, N. Carbone, E. Dodds, J. Kluger, A. Park, N. Rawlings, C. Suddath, F. Sun, M. Thompson, et al., The 50 best inventions, Time Mag. 28 (2011).