Filtering Trolling Comments through Collective Classification

Jorge de-la-Peña-Sordo, Igor Santos, Iker Pastor-López, and Pablo G. Bringas

S^{3}Lab, DeustoTech Computing, University of Deusto, Bilbao, Spain
{jorge.delapenya,isantos,iker.pastor,pablo.garcia.bringas}@deusto.es

Abstract. Nowadays, users are increasing their participation in the Internet and, particularly, in social news websites. In these webs, users can comment diverse stories or other users' comments. In this paper we propose a new method based for filtering trolling comments. To this end, we extract several features from the text of the comments, specifically, we use a combination of statistical, syntactic and opinion features. These features are used to train several machine learning techniques. Since the number of comments is very high and the process of labelling tedious, we use a collective learning approach to reduce the labelling efforts of classic supervised approaches. We validate our approach with data from 'Menéame', a popular Spanish social news site.

Keywords: information filtering, spam detection, web categorisation, content filtering, machine-learning.

1 Introduction

With the appearance of web 2.0 [1], the Internet Community became more sensitive about the primordial users' needs when surfing the net. Since then, the users' dynamic interaction and collaboration was drastically enhanced, and the development of the social networking sites, wikis or blogs, amongst others, started. Social news websites such as Digg[1] or 'Menéame'[2] are very popular among users. These sites work in a very simple and intuitive way: users submit their links to stories online, and other users of these systems rate them by voting. The most voted stories appear, finally, in the front-page [2].

In our previous work [3], we proposed an approach able to automatically categorise comments in these social news sites using supervised machine-learning algorithms. Nevertheless, supervised learning requires a high number of labelled data for each of the classes (i.e., trolling or normal comment). It is quite difficult to label this amount of data for a real-world problem such as the web mining. To generate this information, a time-consuming process of analysis is mandatory and, in the process, some comments may avoid filtering.

Collective classification [4] is a semi-supervised approach that employs the relational structure of labelled and unlabelled datasets combination to increase the

[1] http://digg.com/

[2] http://meneame.net/

J. Lopez, X. Huang, and R. Sandhu (Eds.): NSS 2013, LNCS 7873, pp. 707–713, 2013.

accuracy of the classification. With these relational models, the predicted label will be influenced by the labels of related samples. The techniques of collective and semi-supervised learning have been implemented satisfactorily in fields of computer science like text classification [4], malware detection [5] or spam filtering [6].

Considering this background, we present a novel text categorisation approach based on collective classification techniques to optimise classification performance when filtering controversial comments. This method employs a combination of statistical, syntactic and opinion features of the comments to represent them. Our main contributions are: (i) a new method to represent comments in social news websites, (ii) an adaptation of the collective learning approach to comment filtering, and (iii) an empirical validation which shows that our method can maintain high accuracy rates, minimising the effort of labelling.

The remainder of this paper is structured as follows. Section 2 describes the extracted features of the comments. Section 3 describes the experimental procedure and discussed the obtained results. Finally, Section 4 concludes and outlines the avenues of the future work.

2 Description of the Method

'Menéame' is a Spanish social news website, in which news and stories are promoted. It was developed in later 2005 by Ricardo Galli and Benjamín Villoslada and it is currently licensed as free software. We extracted several features from the comments that can be divided into 3 different categories: opinion, statistical and syntactic features.

- **Statistical Features**
  - **Comment body:** We used the information contained in the body of the comment. To represent the comments we have used the Vector Space Model (VSM) [7]. We used the *Term Frequency – Inverse Document Frequency* (TF–IDF) [8] weighting schema and the inverse term frequency idf_i. As the terming schema we have employed two different alternatives: using the word as the term to weigh and n-grams as terms to weigh. An n-gram is the overlapping subsequence of n words from a given comment.
  - **Number of references to the comment (in-degree):** It indicates the number of times the comment has been referenced in other comments of the same news story.
  - **Number of references from the comment (out-degree):** It measures the number of references of the comment to other comments of the same news story.
  - **Number of the comment:** It indicates the oldness of the comment.
  - **Similarity of the comment with the snippet of the news story:** We used the similarity of the VSM of the comment with the model of the snippet of the news story. In particular, we employ the cosine similarity [9].

  - **Number of coincidences** between words in the comment and tags of the news story.
  - **Number of URLs** in the comment body.
- **Syntactic Features** In this category we count the number of words in the different syntactic categories. To this end, we performed a Part-of-Speech tagging using FreeLing[3]. The following features were used, all of them expressed in numerical values extracted from the comment body: adjectives, numbers, dates, adverbs, conjunctions, pronouns, punctuation marks, interjections, determinants, abbreviations and verbs.
- **Opinion Features**
  - **Number of positive and negative words:** We employed an external opinion lexicon[4]. Since the lexicon contains English words and 'Menéame' is written in Spanish, we translated them to Spanish.
  - **Number of votes:** The number of positive votes of the comment.
  - ***Karma*:** The *karma* is computed by the website based on the users' votes.

3 Empirical Validation

We gathered comments from 'Menéame' from 5th of April, 2011 to 12th of April, 2011. This dataset of comments comprises one week of stories filled by 9,044 comment instances. We labelled each of the comments in one category into *Normal* and *Controversial*. *Normal* means that the comment is not hurtful or hurting, using ia restrained tone. *Controversial*, on the other hand, refers to a comment seeking to create polemic. Our data was finally formed by 6,857 normal comments and 2,187 controversial comments.

We performed two different procedures to generate the VSM of the comment body: (i) VSM with words and terms and (ii) n-grams with different values of n (n=1, n=2, n=3). Furthermore, we removed every word devoid of meaning in the text, called stop words, (e.g., 'a','the','is') [8]. In both cases, we employed an external stop-word list of Spanish words[5].

To evaluate our approach, we applied k-cross validation with $k = 10$. Next, for each training set, we extracted the most important features for each of the classification types using *Information Gain* (IG) [10], an algorithm that evaluates the relevance of an attribute by measuring the information gain with respect to the class and We removed every feature with an IG value of zero. Since the dataset is not balanced for the different classes, we also applied Synthetic Minority Over-sampling TEchnique (SMOTE) [11] to address unbalanced data.

We then accomplished the learning step using different learning algorithms depending on the specific model, for each fold. We employed the implementations of the collective classification provided by the *Semi-Supervised Learning and*

[3] Available in: `http://www.lsi.upc.edu/~nlp/freeling`

[4] Available in: `http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar`

[5] The list of stop words can be downloaded at `http://paginaspersonales.deusto.es/isantos/resources/stopwords.txt`

Table 1. Results in terms of accuracy, TPR, FPR and AUC of the Controversy Level for Word VSM

Dataset	Accuracy (%)	TPR	FPR	AUC
KNN K = 10	67.14 ± 1.92	0.50 ± 0.04	0.27 ± 0.02	0.66 ± 0.02
Bayes K2	75.93 ± 0.65	0.04 ± 0.01	0.01 ± 0.01	0.64 ± 0.03
Bayes TAN	76.64 ± 0.36	0.05 ± 0.01	0.01 ± 0.01	0.64 ± 0.03
Naïve Bayes	74.13 ± 3.74	0.20 ± 0.11	0.09 ± 0.08	0.62 ± 0.03
SVM: PolyKernel	68.35 ± 2.06	0.59 ± 0.05	0.29 ± 0.03	0.65 ± 0.03
SVM: Norm. PolyKernel	69.53 ± 1.55	0.53 ± 0.03	0.25 ± 0.02	0.64 ± 0.02
SVM: PUK	69.54 ± 1.33	0.52 ± 0.04	0.25 ± 0.02	0.63 ± 0.02
SVM: RBFK	68.34 ± 3.33	0.44 ± 0.03	0.24 ± 0.05	0.60 ± 0.02
J48	71.72 ± 2.06	0.31 ± 0.04	0.15 ± 0.02	0.60 ± 0.04
Random Forest N = 100	77.08 ± 0.94	0.18 ± 0.04	0.04 ± 0.01	0.67 ± 0.03

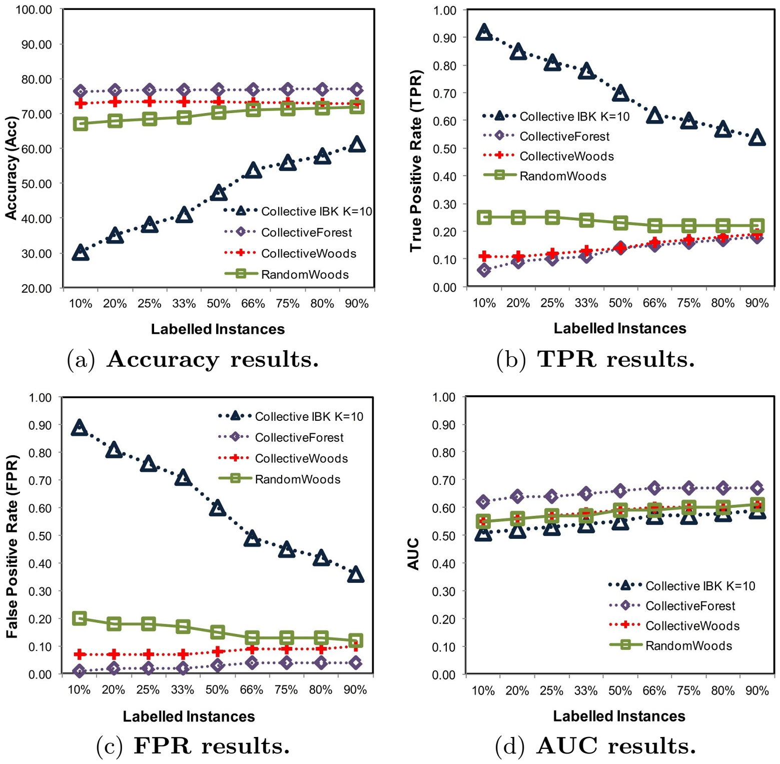
Table 2. Results in terms of accuracy, TPR, FPR and AUC of the Controversy Level for N-gram VSM

Dataset	Accuracy (%)	TPR	FPR	AUC
KNN K = 10	57.32 ± 2.13	0.61 ± 0.05	0.44 ± 0.03	0.63 ± 0.03
Bayes K2	75.60 ± 0.74	0.06 ± 0.02	0.02 ± 0.01	0.65 ± 0.02
Bayes TAN	76.34 ± 0.43	0.06 ± 0.02	0.01 ± 0.00	0.65 ± 0.02
Naïve Bayes	53.81 ± 1.78	0.62 ± 0.02	0.49 ± 0.02	0.59 ± 0.02
SVM: PolyKernel	60.84 ± 1.38	0.74 ± 0.04	0.43 ± 0.01	0.65 ± 0.02
SVM: Norm. PolyKernel	70.72 ± 1.56	0.54 ± 0.05	0.24 ± 0.02	0.65 ± 0.02
SVM: PUK	70.83 ± 1.86	0.49 ± 0.05	0.22 ± 0.02	0.63 ± 0.03
SVM: RBFK	53.42 ± 2.98	0.74 ± 0.03	0.53 ± 0.04	0.60 ± 0.03
J48	71.04 ± 1.54	0.35 ± 0.04	0.17 ± 0.02	0.61 ± 0.02
Random Forest N = 100	76.88 ± 1.30	0.19 ± 0.04	0.05 ± 0.01	0.68 ± 0.03

Collective Classification[6] package for machine-learning tool WEKA [12]. In our experiment approaches, we used the following models: (i) *Collective IBK*, with $k = 10$; (ii) *CollectiveForest*, where the value of the trees to experiment is 100; (iii) *CollectiveWoods*, with 100 trees; and (iv) *RandomWoods*, with 100 trees. In our collective experiments, we examined various configurations of the collective algorithms with different sizes of the $\mathcal{X}$ set of known instances; the latter varied from 10% to 90% of the instances utilised for training (i.e., instances known during the test).

In order to evaluate the contribution of Collective Classification to categorisation comments, we compared the filtering capabilities of our method with some of the most used supervised machine-learning algorithms. Specifically, we used the following models: (i) *Bayesian networks (BN)*, with different structural learning algorithms: K2 and Tree Augmented Naïve (TAN) and a Naïve Bayes Classifier; (ii)Support Vector Machines (SVM), with a polynomial kernel, a normalised polynomial Kernel, a Pearson VII function-based universal kernel (PUK) and a radial basis function (RBF) based kernel; (iii) *K-nearest neighbour (KNN)*, with $k = 10$; and (iv) *Decision Trees (DT)*, trained with J48 (the *Weka* [12] implementation of the *C4.5* algorithm) and Random Forest [13], an ensemble

[6] Available at: `http://www.scms.waikato.ac.nz/ fracpete/projects/collective-classification.`



(a) **Accuracy results.**

(b) **TPR results.**

(c) **FPR results.**

(d) **AUC results.**

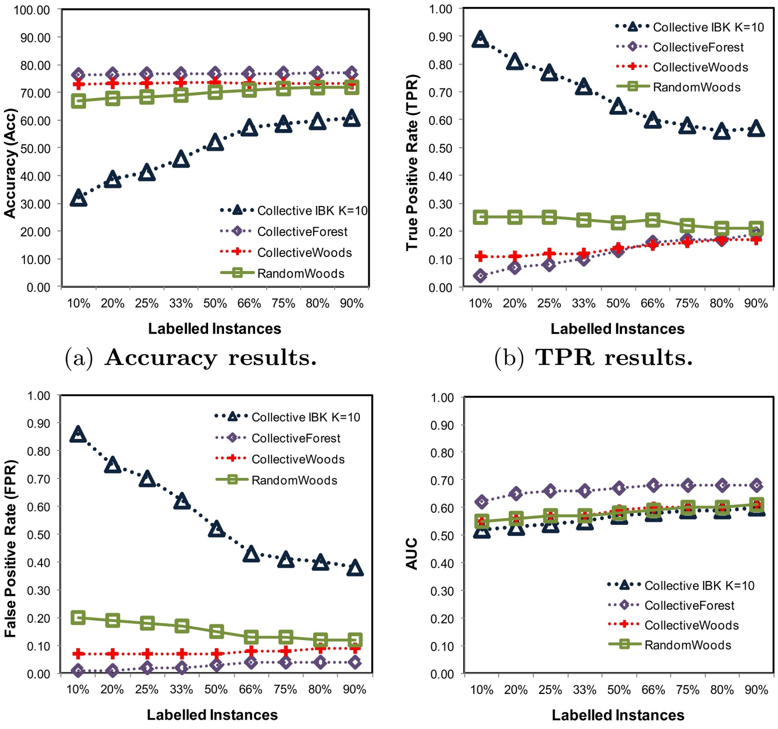
Fig. 1. Results performed with Word VSM features

of randomly constructed decision trees. In particular, we employed $N = 100$ for Random Forest.

Finally, in order to measure the effectiveness of the method, we measured the *True Positive Rate* (TPR) to test our procedure; i.e., the number of the controversial comments correctly detected divided by the total number of controversial comments. We also took in account the *False Positive Rate* (FPR); i.e., the number of normal comments misclassified as controversial divided by the total number of normal comments. In addition, we obtained the *Accuracy*; i.e., the total number of hits of the classifiers divided by the number of instances in the whole dataset. Finally, we recovered the *Area Under the ROC Curve* (AUC), that is computed by plotting the TPR against the FPR under different thresholds and computing the area formed under the generated curve.

Table 1 shows the results with words as tokens using classic supervised learning algorithms, and Table 2 shows the results with n-grams as tokens using classic supervised learning algorithms. Figure 1 shows the results with VSM generated with words, when collective learning algorithm are used, and Figure 2 shows the results with VSM generated with n-grams using collective learning approaches.

Regarding the supervised learning algorithms, Random Forest with N = 100 with words VSM, achieved significant results: 77.08% accuracy, 0.18 TPR, 0.04



(a) **Accuracy results.**

(b) **TPR results.**

(c) **FPR results.**

(d) **AUC results.**

Fig. 2. Results performed with N-gram VSM

FPR and 0.67 AUC. For collective classification, CollectiveForest, using words as terms for the VSM, obtained a accuracy of 76.94% by only labelling the 75% of the dataset, a TPR of 0.16, a FPR of 0.04 and a AUC of 0.67. The results for collective classification are close to the supervised approaches, and the labelling effort has been reduced to 76.94% of the whole data.

4 Conclusions

The problem with supervised learning is that a previous work of comment labelling is required. This process in the field of web filtering can introduce a high performance overhead due to the number of new comments that appear everyday. In this paper, we proposed the first collective-learning-based trolling comment filtering method system that based upon statistical, syntactic and opinion features, is capable of determining when a comment is controversial. We empirically validated our method using a dataset from 'Menéame', showing that our technique, despite having much less labelling requirements, obtains nearly the same accuracy than the best supervised learning approaches.

The avenues of future work are oriented in three main ways. Firstly, we would like to apply additional algorithms to extend the study of filtering trolling comments in social news websites. Secondly, we will incorporate new and different features from the comment dataset to train the models. And finally, we will focus on executing an extended analysis of the effects of the labelled dataset dimension.

References

1. O'Reilly, T.: What is web 2.0: Design patterns and business models for the next generation of software. Communications & Strategies (1), 17 (2007)
2. Lerman, K.: User participation in social media: Digg study. In: Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops, pp. 255–258. IEEE Computer Society (2007)
3. Santos, I., de-la Peña-Sordo, J., Pastor-López, I., Galán-García, P., Bringas, P.: Automatic categorisation of comments in social news websites. Expert Systems with Applications (2012)
4. Neville, J., Jensen, D.: Collective classification with relational dependency networks. In: Proceedings of the Second International Workshop on Multi-Relational Data Mining, pp. 77–91 (2003)
5. Santos, I., Laorden, C., Bringas, P.: Collective classification for unknown malware detection. In: Proceedings of the 6th International Conference on Security and Cryptography (SECRYPT), pp. 251–256 (2011)
6. Laorden, C., Sanz, B., Santos, I., Galán-García, P., Bringas, P.G.: Collective classification for spam filtering. In: Herrero, Á., Corchado, E. (eds.) CISIS 2011. LNCS, vol. 6694, pp. 1–8. Springer, Heidelberg (2011)
7. Baeza-Yates, R.A., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston (1999)
8. Salton, G., McGill, M.: Introduction to modern information retrieval. McGraw-Hill, New York (1983)
9. Tata, S., Patel, J.M.: Estimating the Selectivity of tf-idf based Cosine Similarity Predicates. ACM SIGMOD Record 36(2), 75–80 (2007)
10. Kent, J.: Information gain and a general measure of correlation. Biometrika 70(1), 163–173 (1983)
11. Chawla, N., Bowyer, K., Hall, L., Kegelmeyer, W.: SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research 16(3), 321–357 (2002)
12. Garner, S.: Weka: The Waikato environment for knowledge analysis. In: Proceedings of the 1995 New Zealand Computer Science Research Students Conference, pp. 57–64 (1995)
13. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)