NLP-based Feature Extraction for Automated Tweet Classification

Anna Stavrianou, Caroline Brun, Tomi Silander, Claude Roux

Xerox Research Centre Europe, Meylan, France
Name.surname@xrce.xerox.com

Abstract. Twitter conveys the opinions and interests of people in various topics and domains. In this paper we focus on political data with the purpose of automatically classifying opinion polarities and topic categories. Our experiments show that natural language processing techniques alone do not capture the information wanted. As a result, we use a hybrid method and we propose adding language features in a machine learning process that improved our results.

1 Introduction

Twitter has become one of the most popular micro-blogging services on the Internet, in a few years. Users can send and read tweets which are small text-based messages of up to 140 characters. The limited length of these tweet posts introduces some particularities regarding the text usage.. For instance, abbreviations are often used, orthographic mistakes are made on purpose and hashtags as well as emoticons are present in order to communicate the message of the author in a few words.

Traditional Natural Language Processing (NLP) techniques cannot always deal with these texts that often do not follow even the simplest and most basic syntactic rules. However, they seem to convey information that cannot otherwise be known. In a user profiling task, for example, the gender of the author in certain languages can mainly be identified through NLP techniques. On the other hand, the type of writing in tweet posts is so particular that it has been proven through many experiments that NLP alone is not enough if we want optimal results. As a result, we believe and show, in this paper, that hybrid methods could result in a more efficient analysis of twitter posts. Feeding the natural language analysis output, for example, into a classifier simplifies the learning process.

The work presented in this paper is in the context of a national funded project (ImagiWeb) whose objective is to study the image and reputation of an entity through the social media. An entity can be that of a person (e.g. a politician) or a company. One part of this project focuses on Twitter posts, since the content of such posts often reflects the opinions and comments that the public expresses. A subset of tweets regarding politicians has been manually annotated with two types of categories: a) the opinion polarity (positive, negative and neutral) and b) the topic-category. The topic-category consists in a set of 10 predefined topics that range from ecology to speech capabilities, project proposals and so on. The annotation took place among computer scientists and sociologists. This resulted in an annotation with a low inter-annotator agreement for the topic-category (< 0.4) because people from different domains have a tendency to annotate in different ways. Apart from this issue, that affects our results and makes our task of classification very difficult, another issue is that some of the 10 predefined classes of topic-category have overlapping semantics. For instance, there are classes that refer to the same thing from another point of view (e.g. the period of time – in the future or in the present -) which is difficult to distinguish even manually.

In this paper we focus on the analysis of tweet posts in order to identify their opinion polarities as well as the topic-category to which they belong. Since standard NLP techniques fail to capture the information needed for extracting such information, a combination of NLP and Machine Learning techniques seems more promising for our task. We propose to show that NLP techniques feed classifiers with richer features. Our contributions are on automated tweet classification of political tweets and they can be summarized as: a) Using decomposition of hashtags in the process among other NLP features, b) dealing with a high number of classes (10 classes for topics) and a low inter-annotator agreement.

The paper continues as follows. In the next section we describe related work and the differences it has with our research. In Section 3 we focus on how we have combined NLP and machine learning techniques for the purpose of the specified tweet classification. Section 4 discusses our experiments and Section 5 concludes while presenting future perspectives.

2 Related Work

Techniques of sentiment and topic classification have been extensively applied to text documents. However, the content of the Twitter posts and the style of the text prevent the same exact techniques from being used in the same way. Thus, literature has proposed different approaches for both tasks.

Zhang et al. [21] are using a lexicon based approach to perform sentiment analysis on tweets and they use these prior-annotated data in order to train a sentiment classifier. They do not use NLP features such as hashtags and they do not aim at a supervised classification.

In the opinion mining task, Davidov et al. [8] propose a sentiment analysis by using features such as punctuation and n-grams. The method is supervised and smileys as well as hashtags are used as training labels. They say that using features such as punctuation etc. contributes to the sentiment classification. We do not have the same impression, which is that not all features succeeded in improving our results.

Agarwal et al. [2] also predict opinion polarities on Twitter data. They agree with us that NLP features improve the models. They also use prior knowledge on word polarities and they use a tree representation to combine categories of features as well as a feature-based model. However, our data are different and our task is more difficult due to annotation particularities.

Go et al. [10] are doing sentiment classification on tweets using a 3-class schema. They annotate the data based on the emoticons present in the tweets and then they use distant supervised learning. They do not use NLP features the way we do and their techniques are not hybrid.

In [14] the objective is to relate geographical links with public sentiment. Two methods are presented for sentiment analysis: a dictionary-based one where dictionaries are used to identify polarity and a machine learning one which is supervised following a distant supervision. The methods are compared to each other separately without attempting the hybrid approach.

In [19], they also deal with political data and they use unigram features with a naïve Bayes model. They have a 59% accuracy which is not higher than what our experiments show.

Regarding the supervised learning methods for tweet classification, these are mainly techniques that classify in 2 distinct topics (e.g. [20]). Since this is different from our task, we do not mention these techniques here. For more than 2 classes, an example is in [11] where they define a list of topics and they apply supervised learning using a naïve Bayes multinomial classifier. They are just using bag of words with no semantic overlap in their chosen topics, which makes the task much easier than ours. NLP is not used.

Quercia et al [16] evaluate L-LDA (Labeled LDA) on Twitter with the task of assigning the correct topics to tweet profiles. They claim that this technique has been effective for twitter.

3 Combination of NLP and Machine Learning Techniques

3.1 Context

The objective of the work presented in this paper is to analyze comments posted on Twitter about political entities and perform automatic topic and opinion analysis on these tweets. We aim at creating a model that is able to predict the opinion polarity as well as the topic-category of a tweet post.

Our data include Twitter posts regarding the last French election of May 2012. The posts are in French and they cover a pre-election and post-election period. In this way, at a later stage, we can analyze the evolution of opinions and topics of discussion as they evolve before and after the elections.

Part of the data has been extracted using the Twitter API through adapted keyword queries; this represents about 1,500 users. We checked that none of them did belong to an organization or a company, as we were looking for non-affiliated individuals. We queried the social network with ten French politician names, together with their variations and specified hash tags, in order to extract a wide set of tweets. We have randomly extracted tweets out of this corpus and after preprocessing them we have proceeded with the annotation. 48% of tweets were annotated only once, 46% twice, and 6% three times. We ended up with 5,754 annotated tweets for the polarity task and 6,142 tweets for the topic-category task.

3.2 Syntactic Parsing

Initially we experimented with our syntactic parser [3], which extracts deep syntactic dependencies, from which semantic relations (such as opinions) can be calculated. Deep syntactic analysis consists here in the construction of a set of syntactic relations inspired from dependency grammars [13] [18] - from an input text. These relations link lexical units of the input text and/or more complex syntactic domains that are constructed

during the processing (mainly chunks [1]). These relations are labelled with deep syntactic functions. In addition, the parser calculates more sophisticated and complex relations using derivational morphologic properties, deep syntactic properties (subject and object of infinitives in the context of control verbs), some limited lexical semantic coding (Levin's verb class alternations [12]), and some elements of the FrameNet classification [17].

This parser can, thus, be used as a fundamental component to extract deep syntactic dependencies and compute semantic relations. Having syntactic relations already extracted by a general dependency grammar, we can define our opinion mining system by combining lexical information about word polarities, subcategorization information and syntactic dependencies to extract the required semantic relations. The polarity lexicon has been built from existing resources but also by applying classification techniques over large corpora. The semantic extraction rules are handcrafted [4].

The parser has given very good results on opinion mining tasks when applied to "more" conventional text such as product reviews [4, 5]. Recently it has also performed very well on the Semeval 2014 Aspect Based Sentiment Analysis Task [15], demonstrating performances that go far beyond the task baselines on the different subtasks (see [6]).

However, when applied to Twitter posts, the parser itself does not give very satisfactory results. For instance, on a pre-annotated dataset of 5,700 tweet posts, we noticed 30% accuracy on identifying opinion polarities. As a result, we decided to use a hybrid method and combine the knowledge given by our syntactic parser with a learning model.

3.3 Learning

We have developed a process that analyses tweet posts in order to identify the features that will be used for the classifiers to predict opinion polarities and topic-categories.

We have used the manually annotated set of tweets as our training set. This training set has been pre-processed in order to exclude duplicates and solve issues where annotators did not agree with each other. The corpus has been initially analyzed by our syntactic analyzer. Linguistic information hass been extracted for every tweet in the corpus. We have experimented with different features such as bag of words, bigrams, syntactic categories, information extracted from the hashtags, information regarding negation, opinions, removal of stop-words etc.

For classification purposes, the standard library "liblinear" (http://www.csie.ntu.edu.tw/~cjlin/liblinear/) [9] was used in order to classify tweets according to different categories that have been predefined during the annotation process. We have used logistic regression classifier (with L2-regularization) rather than linear SVM and, thus, in order to calculate probability outputs.

In the multi-class logistic regression, each class c has a separate vector w_c of weights for all the input features. The greater the sum of the features of x, weighted by w_c, the greater the probability of the feature vector x belonging to a class c. More formally,

$$P(c|x; w_c) \propto e^{\sum_{i=1}^{d} w_{ci} x_i}, \quad (1)$$

where x_i is the ith feature (out of d) and the w_{ci} is its weight in the class c. When learning the model using the maximum likelihood method, we try to find the vectors of weight w_c that maximize the product of the class probabilities in the training data. With too many features, such a learning scheme often produces over-fitting. Therefore, it is customary to regularize the model by requiring that the weights cannot on average be too far from zero. One common way to do this is to maximize the regularized log-probability of the training data from which we have subtracted a so called L2-penalty, $L2(w_1, \dots, w_C) = \lambda \sum_{c=1}^{C} \sum_{i=1}^{d} w_{ci}^2$, where C denotes the number of different classes, and the λ is a positive real number that controls the amount of regularization we want. In practice, the λ is often determined using the cross-validation.

Our objective has been to identify the optimal combination of features that yields good prediction results. We have given a lot of importance to avoiding over-fitting, trying to create a non-biased model. We discarded the features with little or no impact over the results, in order to keep the number of features as small as possible. Using less features seemed to have an effect on the classifier. This is probably because the number of features is related to the penalty used by the logistic regression classifier. Therefore, we decided to use a reasonable amount of features. Avoiding overfitting at all costs resulted in our prediction accuracies to be not very high, but we are confident that we would have uniform results for every new tweet dataset that we may have.

Some features we have used for either task of opinion or topic prediction and worth mentioning are:

– Author names:

The use of this feature has resulted in no change in the results. This could be expected because the average number of tweet posts per author is a bit more than 1, so there is not really a topic or polarity distribution per author.

– Bigrams:

The use of bigrams over unigrams did not improve our model significantly so we chose not to use them in our experiments.

– Snippets:

During the annotation process, we kept track of the text snippets that explained why the annotator tagged the post with a particular topic or polarity. This snippet was used as part of the classifier features. It seemed to give better cross-validation results when the features of the snippet were used instead of the features of the full tweet post.

– Date:

The date has been used as a feature because we thought it may be interesting in revealing topic discussions over certain time spans.

– Use of the entity (e.g. the name of a politician):

It was decided not to use this information because we consider that a new, unknown tweet does not necessarily convey this information. We have, however, experimented with this option, but it did not seem to have a considerable effect on our results.

– Hashtags:

Once decomposition techniques have been applied to hashtags, they are, then, analyzed by an opinion detection system that extracts the semantic information they carry. This semantic information can take the form of a feature to feed our classifiers. We give more information about the decomposition of hashtags in the next section.

The extracted information per tweet, in the form of features, is converted to a vectorial format (Euclidean vector) compatible with "liblinear". The respective vectors would be the input format to "liblinear".

3.4 Hashtags

As previously mentioned, we have applied techniques to decompose Hashtags, analyse them and reuse the information extracted for classification purposes. An evaluation of hashtags in short messages have shown that people use different methods to build up these compounds. In many cases, they use uppercase letters to highlight the word boundaries: #IHateCountryMusic. These are the simplest cases, since the task there is simply to detect the uppercase letters and split appropriately "I Hate Country Music". However, if these cases seem simple at a first glance, they are not numerous enough to decide on the proper splitting. The uppercase could belong to a proper name, while the rest of the string is all in lowercase: #IlikeJanescake.

To solve such problems, we applied a method that uses a lexicon and traverses the string twice: from head to tail, then backwards, concatenating letters into words. After each step, we check if the new string is a word against the lexicon. When this is the case, we add it to our buffer. We use a longest match method, which means that we try to produce the longest valid string before pushing it into our buffer. For instance, in #Ihatecountrymusic, the system could consider "count" as being a word, but since we look for the longest match, the system will keep on adding new letters until it reaches the word "country". This algorithm is quite greedy; however it ensures a better recognition rate than a system that would stop every time it finds a match. The evaluation we have performed over 900 hashtags show an F-measure of about 80% [7].

The decomposed hashtags can be used as features for the classifier. In addition, the opinion polarity of these hashtags can also be part of the features. For this purpose, we apply the opinion detection system described in section 3.2 to the list of decomposed hashtags obtained previously from the decomposition step. Here is an example:

#cestridicule (#itsridiculous):

The decomposition steps gives as output: "c est ridicule" and the dependency analysis result gives:

OBJ[PRED](est,ridicule)

OPINION[negative](ridicule,_UNKNOWN-TARGET)

In this example, the system detects a negative sentiment whose predicate is “ridicule”, the target remaining unspecified.

4 Evaluation

In this section we present the results of the experiments for the topic-category and opinion prediction. We have selected the models using a 10-fold cross validation in the training data and evaluated them by their accuracy in the test data.

For the topic-category task, we have a dataset of 6,142 tweets, which we split into 4,913 tweets used for training and 1,229 tweets used for test. We remind here that we had a less than 0.4 inter-annotator agreement, which shows the difficulty of the task.

Table 1. Cross-validation and prediction accuracy results for the topic-category.

Features	Cross-Validation	Prediction
NLP features	44.38	29.37
NLP features + merging of semantically similar classes	48.91	34.17

Table 1. shows the average cross validation results as well as the prediction accuracy for the topic-category classification. This table shows the results of a classifier that has used the NLP features described in the previous section, as well as the results when some semantic merging of classes takes place. The semantic merging consists in reducing the number of classes. As mentioned previously, the topic-category consists of 10 predefined classes. Some of these classes have a semantic relationship. Thus, their merging can reduce the total number of classes from 10 to 7 and this, apparently, gives a better result. This is not surprising, of course, because when the number of classes reduces, the classifier is more capable to differentiate between them. Still, the prediction accuracy is quite low. We will see later on how we could improve this. In the training data, the class with the highest presence is present with a 29.6% distribution. Assigning this class to the test data gives approximately a 30% accuracy, a result that shows that in the second case our model learns a little bit.

For our opinion polarity task we have 5,754 tweets split in 4,602 tweets for the training set and 1,229 tweets for the test set. The inter-annotator agreement for polarity was higher (around 0.8).

Table 2. Cross-validation and prediction accuracy results for the opinion polarities.

Features	Cross-Validation	Prediction
NLP features	61.28	56.77
NLP features + syntactic analysis of opinion	62.13	56.6
NLP features over the extract	66.41	61.2
NLP features over the extract + Date	66.88	60.85
NLP features over the extract + date + syntactic analysis of opinion	67.99	61.46

Table 2. shows the results regarding the opinion polarity. In this case, we have experimented not only with the NLP features extracted from the tweet post but also with NLP features extracted from the ‘snippet’. The snippet is part of the annotated data and it specifies the part of the tweet that explains why the annotator tagged the tweet as positive, negative or neutral. The “syntactic analysis of opinion” mentioned in the table is the opinion tag given from our opinion mining system. When this is integrated into the system, the results are improved. Using the extract as well as the information about the opinion has not been mentioned in the topic-category classification because in our case, with our data, they do not improve the results.

In the training data, the class with the highest presence is the negative-polarity one and is present with a 54% distribution. Assigning this class to the test data gives approximately a 56% accuracy, a result that again shows that our model learns a little more in some cases.

There has been an effort to improve the topic classification because the task is very difficult and the presence of 10 classes with different distributions is not an easy task to perform. As a result, we have experimented with binary classification to see if the topic-category results are improved.

In order to proceed with the binary classification, we have identified the distribution of the 10 different classes present in the training corpus.

The class with the highest distribution is about 30% while the second class is present with a 13% distribution. Therefore, we have selected the class with the highest presence to be the first class for which we will perform binary classification. The training corpus is annotated with CLASS_1 and NOT_CLASS_1 tags and the model is fitted. Once we have this model, we remove the instances of CLASS_1 and we do similarly for CLASS_2. For the binary classification of both of these classes, we performed resampling with replacement because the data were unbalanced and this was causing difficulties to the model to learn predicting both classes.

In general, we created a model for the prediction of CLASS_1, the prediction of CLASS_2 and a final model for the prediction of the rest of the 8 classes. Having 3 models in total for the prediction meant developing an algorithm that allows for the application on these 3 different models on the test dataset. The results are given in Table 3.

Table 3. Binary classification results for topic-category.

	Cross-Validation	**Prediction**
CLASS_1/NOT_CLASS_1	85.28	62.57
CLASS_2/NOT_CLASS_2 (after removal of CLASS_1 instances)	92.10	68.42
The rest of the classes (after removal of CLASS_2 instances)	49.58	38.24

Table 3 shows that prediction accuracy for the first three models is quite high compared to the results of Table 1. In order to have some more comparable results we merge these three models. This gives a prediction accuracy of 40.03% which is higher than the maximum prediction accuracy of Table 1. Splitting the third model to more models could probably increase the accuracy, but the problem remains the same: annotation on this political data is very special, the classes are many and their semantics are sometimes not clear to distinguish even for a human. Thus, the 40.03% accuracy is considered to be a good result.

5 Conclusion and Perspectives

In this paper we have attempted to provide a model that predicts the opinion and a specific predefined topic for a tweet in the French political context. This is a hard task whose difficulty increases with the number of classes defined and it seems to be very much dependent on the annotations. In our case, annotation was done by sociologists as well as computer scientists and we have noticed a difference in the way of annotating, which affects highly our results.

Our objective is to keep improving our models. More research around feature analysis will be carried out, and more experiments with different features will be made. We also plan to add more features yielded by our syntactic analyzer such as POS tags, or tense (past, future, etc.) because some of our classes are dependent on that e.g. the speech of a politician about what has happened in the past versus the speech for his or her future proposals.

We have also extracted the words that usually appear per class. In the near future we plan to identify possible word distributions that will help us tune our models.

Another issue of which we should think, is that of the annotation. The annotation part should probably be re-organized especially in the task of topic-category. A multiple-class labelling is probably more interesting. Instead of asking the annotator to give a single class as an answer, we could ask her to give a class ranking based on the most relevant class. This could improve the models and our results.

6 Acknowledgements

This work was partially funded by the project ImagiWeb ANR-2012-CORD-002-01.

References

1. Abney, S.: Parsing by chunks. In Principle-Based Parsing: Computation and Psycholinguistics. Boston:Kluwer Academic Publishers, 1991.
2. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.: Sentiment Analysis of Twitter Data. In Proceedings of the Workshop on Language in Social Media (LSM 2011), pages 30–38, 2011.
3. Ait-Mokthar, S., Chanod, J.P. : Robustness beyond Shallowness: Incremental Dependency Parsing. Special Issue of NLE Journal, 2002.
4. Brun, C.: Detecting opinions using deep syntactic analysis. In RANLP'2011, Hissar, Bulgaria, pages 392–398, 2011.
5. Brun, C.: Learning opinionated patterns for contextual opinion detection. In COLING'2012, Mumbai, India, pages 165–174, 2012.
6. Brun, C., Popa, D., Roux, C.: XRCE: Hybrid Classification for Aspect-based Sentiment Analysis. To appear in International Workshop on Semantic Evaluation (SemEval), Dublin, Ireland, August 2014.
7. Brun, C., Roux, C.: Décomposition des « hash tags » pour l'amélioration de la classification en polarité des « tweets ». In Proceedings of 21ème Traitement Automatique des Langues Naturelles, Marseille, July, 2014.
8. Davidov, D., Tsur, O., Rappoport, A.: Enhanced sentiment learning using Twitter hashtags and smileys. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters (COLING '10). Association for Computational Linguistics, Stroudsburg, PA, USA, 241-249, 2010.
9. Fan, R-E., Chang, K-W., Hsieh, C-J., Wang, X-R., Lin, C-J.: LIBLINEAR: A library for large linear classification. In Journal of Machine Learning Research 9 (2008) 1871-1874, 2008.
10. Go, A., Bhayani, R., Huang, L.: Twitter Sentiment Classification using Distant Supervision, technical report (http://help.sentiment140.com/)
11. Lee, K., Palsetia, D., Narayanan, R., Patwary, Md. M.A., Agrawal, A., Choudhary, A.: Twitter Trending Topic Classification. In Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops, 2011.
12. Levin, B.: English Verb Classes and Alternations A Preliminary Investigation. University of Chicago Press, Chicago and London, 1993.
13. Mel′čuk, I.: Dependency syntax: theory and practice. State University of New York Press, 1988.
14. Nguyen, V.D., Varghese, B., Barker, A., The Royal Birth of 2013: Analysing and Visualising Public Sentiment in the UK Using Twitter, In , 2013.
15. Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.: Semeval-2014 task 4: Aspect based sentiment analysis. To appear in International Workshop on Semantic Evaluation (SemEval), Dublin, Ireland, August, 2014.
16. Quercia, D., Askham, H., Crowcroft, J.: TweetLDA: supervised topic classification and link prediction in Twitter. In WebSci, page 247-250. ACM, 2012.
17. Ruppenhofer, J., Ellsworth, M., Petruck, M.R.L., Johnson, C.R., Scheffczyk., J.:.FrameNet II: Extended theory and practice. Technical report, ICSI, 2005.
18. Tesnière, L.: Eléments de syntaxe structurale. Klincksiek eds., 1959.
19. Wang, H., Can, D., Kazemzadeh, A., Bar, F., Narayanan, S.: A System for Real-time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 115–120, 2012.
20. Yerva, S. R., Miklos, Z., Aberer, K.:"What have fruits to do with technology?: the case of orange, blackberry and apple," in Proceedings of the International Conference on Web Intelligence, Mining and Semantics , 2011.
21. Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., Liu, B.: Combining Lexicon-based and Learning-based methods for twitter sentiment analysis. In Technical report, HP Laboratories, 2011.