SentimentalITsts at SemEval-2016 Task 4: building a Twitter sentiment analyzer in your backyard

Cosmin Florean, Oana Bejenaru, Eduard Apostol, Octavian Ciobanu, Adrian Iftene and Diana Trandabăţ
University "Alexandru Ioan Cuza" of Iaşi, Romania
{cosmin.florean, oana.bejenaru, eduard.apostol, octavian.ciobanu, adiftene, dtrandabat}@info.uaic.ro

Abstract

The paper presents the system developed by the *SentimentalITsts* team for the participation in Semeval-2016 task 4, in the subtasks A, B and C. The developed system uses off the shelf solutions for the development of a quick sentiment analyzer for tweets. However, the lack of any syntactic or semantic information resulted in performances lower than those of other teams.

1 Introduction

Slowly but surely, social media replaced the traditional sources of information: people's need to be constantly updated changed our behavior from buying a newspaper or watching TV, to using a Facebook or Twitter account to visualize, in a customizable manner, the day's hottest news, with the bonus of being able to also comment on them.

Social media sites gained their popularity due to the "freedom" of expression they induce in people's mind: being able to post real time messages about your opinions on whatever topic you come across, discuss political and social decisions, complain, express gratitude or exchange impressions about products you use in everyday life.

Texts shared through social media applications offer us the information that we need: for example, the reviews of a product provide us useful information about its advantages and disadvantages, while the text of an advertisement invites us to eat at the new Chinese restaurant in town.

As huge amounts of texts become available through social media, a challenging task concerns the organization and processing of this information to extract knowledge. Natural language processing tools trained on large news corpora have usually problems when applied to unstandardized social media inputs, mainly due to the fact that social media content can appear in various forms (Becker et al., 2012), from photos and video updates to news, offers and literary works, and various informal formats.

Twitter is micro-blogging platform where people can send messages to one or multiple users, follow friends and read messages without much difficulty. Twitter messages, commonly known as tweets, are limited to 140 characters, and frequently include hashtags (labels which should make it easier for users to find messages with similar content), all in one making Twitter analysis charming.

Out of the 5 subtasks of Semeval-2016 task 4, the *SentimentalITsts* participated in subtask A: Message Polarity Classification , subtask B Tweet classification according to a two-point scale and subtask C Tweet classification according to a five-point scale. Subtask A asked to classify a given tweet in three categories: positive, negative or neutral, according to the identified sentiment (Nakov et al., 2016). The tweet was known to be about a specific topic (by topic is meant anything people usually express opinions about on social networks: a product, a political candidate, a policy, an event, etc.) and the topic was given by the task organizers. Subtask B is a two-scale sentiment classification task, where tweets need to be identified as

Proceedings of SemEval-2016, pages 248–251,
San Diego, California, June 16-17, 2016.

positive or negative. Subtask C is a refinement of the previous subtasks, demanding a five-point scale: very positive, positive, neutral, negative, very negative. A five-point scale is now ubiquitous in the corporate world where human ratings are involved; e.g., Amazon, TripAdvisor, Yelp, and many others, all use a five-point scale for their reviews.

The remaining of this paper is structured as follow: Section 2 provided an overview of the state of the art applications the team has considered for sentiment analysis of social media, Section 3 presents the method used by the *SentimentalITsts* to develop their own sentiment analyzer, Section 4 offers some not official results used in analyzing the system's performance, before the final section drawing conclusions and further directions.

2 State of the art

Specific processing tools (such as POS taggers or anaphora resolution systems), score a higher performance if used on the same text type as the ones they were trained on. In other words, we will have better results if using a POS tagger trained on news corpora to analyze news texts, rather than speech transcripts.

Thus, the short dimension of tweets and their creative informal spelling have raised a new set of challenges to the natural language processing field. How to handle such challenges so as to automatically mine and understand the opinions and sentiments that people are communicating has been the subject of several research (Jansen et al., 2009; Barbosa and Feng, 2010; Bifet and Frank, 2010; Davidov et al., 2010; O'Connor et al., 2010; Pak and Paroubek, 2010; Tumasjen et al., 2010; Kouloumpis et al., 2011; Russell 2013; Pang et el., 2002).

A list of functional applications developed until now on Sentiment Analysis and API's that have a great success over the internet is presented below:

Sentiment140[1] (formerly known as "Twitter Sentiment") allows the discovery of the sentiment associated to a brand, product, or topic on Twitter. The API (Go et al., 2009) uses a maximum entropy classifier, trained on a set of automatically extracted tweets. The training corpus of 1.600.000 tweets is created relying on the use of emoticons (tweets with happy smileys suggest a positive contents, while tweets with sad/anger smileys refer to negative contents). The API lets users classify tweets and integrate sentiment analysis functionality into their own websites or applications, using RESTful calls and responses formatted in JSON.

Werfamous[2] is another webservice offering a sentiment search ability for a user selected term.

Sentiment Analysis with Python NLTK Text Classification***:*** It can classify the text introduced on one of three groups: positive, negative or neutral. Using hierarchical classification neutrality is determined first, and sentiment polarity is determined second, but only if the text is not neutral. The NLTK Trainer is used to train classifiers for the sentiments based on twitter sentiment or movie reviews. NLTK[3] (Bird et al., 2009) is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, etc.

DatumBox[4]: an OpenSource API that allows users to access the web services offered by DatumBox. This services include Sentiment Analysis on any post using a 3 point scale considering that the topic of the post is given.

AlchemyAPI[5] (Turian, 2013) launched in 2009, is a company that uses machine learning (specifically deep learning) to do natural language processing (specifically semantic text analysis including sentiment analysis) and computer vision (face detection and recognition) for its clients both over the cloud and on-premises.

LexAlytics[6] is a web platform for media monitoring, offering nice visualization tools and powerful document processing capabilities.

For the Semeval-2016 participation, the *SentimentalITsts* team has used a self-trained Naive

[1] http://help.sentiment140.com/home

[2] http://werfamous.com/
[3] http://www.nltk.org/
[4] http://www.datumbox.com/
[5] http://www.alchemyapi.com/
[6] https://www.lexalytics.com/

Bayes classifiers, combined with the existing Alchemy-API for the cases where the classifier's output score was below an empirically established threshold.

3 Architecture

Building a Social Media Monitoring tool requires at least 2 modules: one that evaluates how many people are influenced by the campaign and one that finds out what people think about the brand.

For the second tool, being able to evaluate the opinion of the users is not a trivial matter. Evaluating their opinions requires performing Sentiment Analysis, which is the task of automatically identifying the polarity, the subjectivity and the emotional states of a particular document or sentence. It requires Machine Learning and Natural Language Processing techniques.

The reminder of this section will present a short description of the classes and objects used for the development of the three systems which participated in SemEval subtasks A, B and C. The architecture for the three systems was similar, the main difference being the way Naïve Bayes classifiers were trained: for 2, 3 or 5 classes, respectively. The training instances were obtained from the train and development corpora offered by the organizers of the SemEval-2016 task, and the internal evaluation was performed on the test development data.

- NaiveBayes Class
  ○ main part of the Text Classifier
  ○ implements methods such as train() and predict() that are responsible for training a classifier and using it for predictions
  ○ use external methods to preprocess and tokenize the document before training

- NaiveBayesKnowledgeBase Object
  ○ output of training which stores all the necessary information and probabilities used by the classifier

- Document Object
  ○ training and prediction texts in the implementation are internally stored as Document Objects
  ○ stores all the tokens of the document, their statistics and target classification of the document

- FeatureStats Object
  ○ stores several statistics that are generated during the Feature Extraction phase.

- FeatureExtraction Class
  ○ calculates internally several of the statistics that are actually required by the classification algorithm in the later stage, and all these stats are cached and returned in a FeatureStats Object to avoid their recalculation.

- TextTokenizer Class
  ○ simple text tokenization class, responsible for preprocessing: cleaning and tokenizing the original texts, removing special symbols, identifying and annotating hashtags and smileys, standardizing word with repeated letters, and converting them into Document objects.

Similar to (Go et al, 2009 and Pang et al., 2002), the Naïve Bayes classifiers were trained using the following features: tokenized unigrams, emoticons, hashtags.

4 Non-official error analysis

In order to check the system's weakness and straightness, an internal evaluation was performed before the official evaluation, for each substask.

When analyzing the errors found in the classification for subtask A (Fig.1), one can easily note that the system is positive-biased, i.e. it gave too many positive answers. Thus, out of the 29% of negative instances wrongly classified, 77% were classified as positive, while 23 as neutral. Similarly, for the neutral instances in gold which were misclassified, 89% were identified as positive, and 11% as negative.

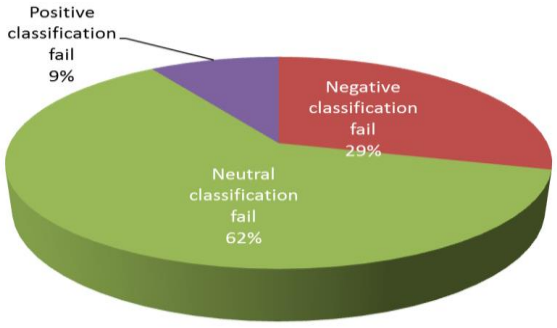

Figure 1. Internal evaluation for three-point scale error cases

For subtask B, the most misclassified category was the negative one. Table 1 presents the confusion matrix for the two categories. It is worth noticing that the system developed for subtask B is significantly better than the one for subtask A, according to our internal evaluation.

	Negative	Positive
Negative	81,14	18,86
Positive	3,40	96,60

Table 1. Confusion matrix for the five-scale task, evaluated on test development data

For subtask C, the system was biased towards the neutral classification. Thus, in case of doubt or when no other classification goes beyond a confidence score, the neutral classification was selected. The error matrix is presented in table 2.

	Very neg.	**Very pos.**	**Pos.**	**Neutr.**	**Neg.**
Very neg.	88,31	0,00	6,68	5,01	0,00
Very pos.	0,00	65,10	32,32	2,29	0,29
Pos.	4,39	0,00	62,42	30,72	2,47
Neutr.	0,00	0,00	62,14	36,27	1,59
Neg.	0,00	0,00	32,73	12,90	54,37

Table 2. Confusion matrix for the five-scale task, evaluated on test development data

5 Official evaluation and discussions

The official evaluation placed the *SentimentalITsts* on the 32nd place for subtask A (see table 3 for details of performances on each dataset). For subtask B, the team was placed 15th out of 19th, and for subtask C the official classification was at rank 9 out of 11 (see for details Nakov et al., 2016).

SentimentalITsts	2013		2014				
	Tweet	SMS	Tweet	Tweet sarcsm	Live Journal	2015 tweet	2016 Tweet
	0.33	0.23	0.39	0.28	0.32	0.34	0.33

Table 3. Official results for the SentimentalITsts team, task A

In this version of the system, no syntactic or semantic information was used. Similarly, hashtags or smileys, although they seemed useful in initial tests, ultimately showed that they bring more noise than relevant information, and were thus removed from the message files.

References

Barbosa, L. and Feng, J. (2010). *Robust sentiment detection on twitter from biased and noisy data*. Proceedings of Coling 2010 .

Bifet, A. and Frank, E. (2010). *Sentiment knowledge discovery in twitter streaming data*. Proceedings of 14th Int. Conference on Discovery Science.

Bird Steven, Klein E., Loper E., (2009) *Natural Language Processing with Python*, O'Reilly Media.

Davidov, D., Tsur, O., Rappoport, A. (2010). *Enhanced sentiment learning using twitter hashtags and smileys*. Proceedings of Coling 2010.

Go A., Bhayani R., Huang. L. (2009) Twitter Sentiment Classification using Distant Supervision, *Technical Report*.

Jansen, B.J., Zhang, M., Sobel, K., Chowdury, A. (2009). *Twitter power: Tweets as electronic word of mouth*. Journal of the American Society for Information Science and Technology 60(11):2169-2188.

Kouloumpis, E., Wilson, T., and Moore, J. (2011). *Twitter Sentiment Analysis: The Good the Bad and the OMG*! Proceedings of ICWSM. 2011

MA Russell (2013) Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More.

Nakov P., Ritter A., Rosenthal S., Stoyanov V., Sebastiani F.(2016) *SemEval-2016 Task 4: Sentiment Analysis in Twitter*, Proc. of SemEval '16.

O'Connor, B., Balasubramanyan, R., Routledge, B., and Smith, N. 2010. From tweets to polls: Linking text sentiment to public opinion time series. Proceedings of ICWSM.

Pak, A. and Paroubek, P. 2010. *Twitter as a corpus for sentiment analysis and opinion mining*. Proceedings of LREC 2010.

Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. (2002) "Thumbs up?: sentiment classification using machine learning techniques." Proceedings of EMNLP, pp. 79-86.

Tumasjan, A., Sprenger, T.O., Sandner, P., Welpe, I. (2010). Predicting elections with twitter: What 140 characters reveal about political sentiment. Proceedings of ICWSM 2010.

Turian Joseph, (2013) *Using AlchemyAPI for Enterprise-Grade Text Analysis*, PhD Thesis.