Gradiant-Analytics: Training Polarity Shifters with CRFs for Message Level Polarity Detection

Héctor Cerezo-Costas, Diego Celix-Salgado
Gradiant - Galician Research and Development Center in Advanced Telecommunications
Edificio CITEXVI, local 14
Vigo, Pontevedra 36310, SPA
{hcerezo, dcelix}@gradiant.org

Abstract

In this paper we present our solution for obtaining sentiment at message-level of short sentences. The system combines the use of polarity dictionaries and *Conditional Random Fields* to obtain syntactic and semantic features, which are afterwards fed to a statistical classifier in order to obtain the sentence polarity. To improve results, an ensemble of classifiers was employed by combining the individual outputs with majority voting strategy. Our solution was evaluated in the SemEval 2015 Task 10, subtask B: Sentiment Analysis in Twitter, achieving competitive performance in all testsets.

1 Introduction

Sentiment Analysis (SA) is a hot-topic in the academic world, and also in the industry. In SA, a label is automatically assigned to a piece of content carrying the polarity of the composition. The relevance for the web industry is clear, as new services promote content sharing among users. The number of registers generated by these services is paramount, discouraging manual analysis. Hence, automatic systems capable of processing this information have great value for the industry. Many services, such as web advertisement, recommendation, and mail campaigns (to name a few) could benefit from the information gathered with polarity analysis of user content.

This work is focused on message-level sentiment analysis, that is, the objective is the assignment of polarity to a small piece a text, typically one or two sentences with less than 140 characters. This restriction is motivated by the popularity of microblogging services such as Twitter. Here, users write messages of up to 140 characters to share information, their opinions or their feelings with other users. Those messages are shared in real time, and are a sample of the public opinion. Therefore, these small compositions published in microblogging sites can be analyzed to deduce the opinions about any topic of interest.

Nevertheless, automatic systems are not perfect. The results of the sentiment analysis in short sentences is not completely reliable. State of the art solutions are still far from being comparable to human performance, though very promising results were obtained recently using deep learning systems (Socher et al., 2013; Tang et al., 2014), and a careful selection of features with *Support Vector Machines* (SVM) (Zhu et al., 2014) or other statistical classifiers (Go et al., 2009).

This paper describes our sentiment classifier for short sentences and the results in our participation in the SemEval 2015 competition. We have implemented a supervised solution for learning the polarity of short messages. We made extensive use of sequential *Conditional Random Fields* (CRFs) in order to obtain the scope of polarity modifiers and shifters (e.g. negation, intensification). Although a complete explanation of CRFs is out of scope of this paper, the reader can obtain comprehensive information about it in the literature (Lafferty et al., 2001; Sutton and McCallum, 2011).

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 539–544,
Denver, Colorado, June 4-5, 2015.

2 The SA System

This section presents a detailed overview of our system for sentiment tagging of short sentences. Our system performs several steps over each register to determine the polarity of the sentence. Initially, each register is preprocessed to obtain a normalized representation of the data. Next, syntactic information is extracted generating high-level features. As a final step, the features extracted in previous analysis are fed to a statistical classifier, obtaining the polarity of the register.

This is a supervised system, and therefore it needs a learning phase where data are tagged manually. The supervised models are trained only with the data provided by the organization, and therefore it can be considered a constraint solution.

2.1 Preprocessing

The sort of language used in microblogging services is colloquial style, with misspelled words and grammatical and syntactic errors. In order to solve this problem, basic normalization is performed as the first step. The actions executed in this stage are the substitution of emoticons for equivalent words and the substitution of frequent abbreviations. By lack of space, a complete Lookup Table of emoticons cannot be displayed in this paper but a sample of relevant transformations are in Table 1. We divided the emoticons in twelve categories: angry, bad, boring, complicity, happy, laugh, love, neutral, sad shy, surprise and worried.

One kind of specific language artefacts appearing in Twitter are hashtags. Hashtags are small pieces of text which usually contain valuable information to extract the sense of a whole sentence. Users use those chunks to voice those parts more relevant of the message and, very frequently, they are opinionated. Nevertheless, hashtags do not follow the grammar rules (e.g. no case used, words are stuck together, incomplete sentences without subject, verb, etc). To deal with the multiword problem of hashtags, we developed a module that uses CRFs with character-level features to find word terminations in hashtags. If more than one word is found, the system handles them as separated words in following steps.

Table 2 contains different multi-word hashtags that appear in the testsets provided in the SemEval competition and the corresponding output of the hashtag splitter. One of the hashtags, #whatabadass, is wrongly processed but with no significant change in meaning. In internal tests, 93% of words are correctly extracted by this approach.

Emoticons	Replacement
:), :-), :o), etc	happy
XD, x-D, xD, etc	laugh
:*, :^ *, etc	love
;), ;-), ;D, etc	complicity
:(, :'-(, :-[, etc	sad
D;, DX, D:, etc	worried
:@, :-||, etc	angry
o_O, o.O, o_0, etc	surprise
:O, >:O, :-O, etc	boring
:-###.., etc	bad
:$, ^^;, etc	shy
:#, :-#, :X, etc	neutral

Table 1: Sample of Emoticon Transformations.

Input Hashtags	Output Words
#classicmovielotto	classic movie lotto
#notupinhere	not up in here
#Thatisall	That is all
#shoptilwedrop	shop til we drop
#whatabadass	what a bad ass
#wordtomymuva	word to my muva

Table 2: Inputs and Outputs of the Hashtag Splitter.

2.2 Word Features

Our system uses several dictionaries as an input for different steps of the feature extraction process. These dictionaries are used to extract labels that get combined with features in the learning steps.

- Polar Dictionary: contains polar words, positive and negative, in English. This is a general purpose dictionary and no adaptation to the context of analysis was performed. If a word is registered as a positive/negative word, it is labelled with the corresponding polar tag. In case of ambiguity (the word appears in both dictionaries) the polarity label is not used for this word. The baseline for this dictionary was SentiWordNet (Baccianella et al., 2010), aug-

mented after observation of training records.

- Denier particles: this dictionary contains particles that reverse the polarity of the words affected by them (e.g. not, no, etc). Detecting the scope of negation plays an important role in detecting the polarity of a sentence. The academic literature follows different approaches, such as hand-crafted rules (Sidorov et al., 2013; Pang et al., 2002; Athar, 2011), or CRFs (Lapponi et al., 2012b; Nakagawa et al., 2010; Councill et al., 2010).

- Reversal verbs: their behaviour is similar to denier particles. Some verbs reverse the polarity of the content under their scope of influence (e.g. *avoid*, *prevent*, *solve*, etc). In order to obtain the list of reversal verbs, basic syntactic rules and a manual supervision was applied afterwards. A similar approach can be found in (Choi and Cardie, 2008).

- Comparators and Superlatives: a dictionary with comparatives and superlatives was built in a similar way as the polar dictionary. There is a bit of redundancy with this feature as the morphological tagger used by the system gives the same information. However, the syntactic parser is not very reliable for informal style, unless it is specifically trained, which is not the case of our system. This information is needed to track intensification and comparisons within a sentence.

2.3 Syntactic Features

Several language constructions can act like polarity shifters with those parts influenced by them. This is the case of negation particles and some specific verbs. Detecting the scope of this modifiers is a hard task. Our system employs CRFs to obtain labels of those part of sentences that can act as polarity shifters, or that are influenced by polarity shifters. In this sense, we consider the detection of these scopes as an special case of a sequential labelling problem. CRFs are supervised techniques and they learnt the parameters of the system using manually labelled examples. We have built training records using a subset of the data available in the task.

Input Features
words, word bigrams, word trigrams, stems, stem bigrams, stem trigrams, PoS, PoS bigrams, PoS trigrams, distance to denier particle, distance to denier verb, distance to advers. particles

Table 3: Input Features of CRFs.

Our system follows a similar approach to (Lapponi et al., 2012a) but it was enhanced to track intensification, comparisons within a sentence, and the effect of adversative clauses (e.g. sentences with *but* particles). Figure 1 shows an example of the labels assigned by the system to each word of a sentence. Table 3 show the inputs and the combination of features included in the CRFs. The particles of negation (e.g. *none*, *not*), denier verbs (e.g. *prevent*, *avoid*) and others present in internal dictionaries such as *more*, *very*, *less* or *but* are marked as CUEs of negation, intensification and adversarial scopes respectively.

Sentences are tagged to obtain morphosyntactic data to use this information as input to the polarity shifter modules. In our case, we use the Freeling tool (Padró and Stanilovsky, 2012) for this stage. Freeling is an open source suite with tools to analyse textual data. It contains parsers with different degree of complexity but to the purpose of our system, only the *Part of Speech (PoS)* information was needed. We do not use dependence parsing as input feature in contrast to previous state of the art. The approach followed could experience problems with discontinuous scopes (e.g. when subordinate or participle clauses are intermingled within a sentence), but this problem is negligible due to the typically direct and colloquial style of short sentences.

The labels gathered with the CRF modules are used in conjunction with the Word, Stem, PoS and polar dictionaries to generate high-level features which serve as input to the classifier and thus to assign the polarity to the message in the final step.

2.4 Classification Algorithm

All the characteristics from previous steps are included as input features of a statistical classifier. The lexical features (word, stem, word and stem bigrams and flags extracted from the polar dictionaries) with

	Joachim	Rodriguez	may	have	missed	out	on	the	pink	jersey	but	he	is	top	ranking
CRF-INT	O	O	O	O	O	O	O	O	O	O	O	I	I	CUE_I	I
CRF-NEG	N	N	N	N	CUE_N	N	N	N	N	N	O	O	O	O	O
CRF-ADV	O	O	O	O	O	O	O	O	O	O	CUE_A	A	A	A	A

Figure 1: Example sentence with the CRF label notation.

Polarity	Positives	Neutral	Negatives
N. Samples	3456	4468	1432

Table 4: Training vector.

PoS and the labels from the CRFs. The algorithm employed for learning was a logistic regressor. Due to the size of the feature space and their sparsity, l1 (0.000001) and l2 (0.000001) regularization was applied to learn the important features and discard those with low relevance to the task.

2.5 Ensemble of classifiers

It could be possible to use the whole training vector available in one unique classifier, but we chose a different strategy that provided better results.

The ensemble was obtained by replicating the individual schema but using a small subset of the data available for training. The final decision combines the outputs of the classifiers using majority voting. Despite the time complexity of the ensemble and the lower precision of the individual classifiers, this strategy yielded better results than the one-classifier approach (between 1% and 3%) though it depended on the individual execution. An ensemble of 15 to 30 classifiers performed reasonably well in the evaluation tests.

3 Evaluation

3.1 Dataset

To train and validate the system during development, the SemEval organization provides the team competitors with a) an index set of tweets (that should be downloaded by teams), and b) a progress and input test for fair comparison of the different approaches. All the records that can be used as training vector are labelled with a tag (positive, negative and neutral). Due to cancellations of tweets that were not available, our system employed a subset of training of those provided by the organization. Table 4 shows the distribution of the training vector used by our system. A subset of those records are employed to train the CRF models.

Test	**F-score**
LiveJournal 2014	72.63
SMS 2013	61.97
Twitter 2013	65.29
Twitter 2014	66.87
Twitter 2014 sarcasm	59.11
Twitter 2015	60.62
Twitter 2015 sarcasm	56.45

Table 5: Performance in progress and input test.

Finally, the performance of our system is evaluated using a F-score that combines the F-score of positive and negative tweet, whilst neutral records are used to reckon the precision and recall of the positive/negative classes. We refer the reader to (Rosenthal et al., 2015) for a complete description of the task and the evaluation process.

3.2 Results

Table 5 shows the results of our system in the progress test of 2014 and the new input tests of 2015. The system shows a distinguished performance in nearly all the progress tests of 2014. It achieved 17th position in Twitter 2014 sentences, 1st in Twit-

ter Sarcasm and 11th in LiveJournal2014. In SMS 2013 and in Twitter 2013 datasets we achieved also a good result (21th and 22th respectively). Regarding sarcasm detection in 2014 dataset, our system had good results in tweets with hashtags (25 right answers out of 35) whereas it was more prone to fail when users expressed positive opinions over negative events. Paying more attention to these specific constructions would lead to better results in the future.

In the new tests of Twitter 2015, our system performed in the 16th position of all competitors in both sarcasm and normal datasets. There is a clear gap of 6 points between the 2014 and 2015 Twitter F-score and the new testset. Our system is supervised and was only trained with the vector provided by the SemEval community which could mean the gap between the training and test vectors has increased this year. In this sense, it would be interesting to train with external records to see if the performance over the 2015 tests could be improved.

4 Conclusions

This paper shows the solution developed by Gradiant (http://www.gradiant.org) for the Sentiment Analysis Task 10 (subtask B) of SemEval 2015. The system finished in a notable 16th position out of 40 participants. In general terms, our system exhibits stability in all the different subtasks, achieving the 1st position in one of them, 2014 Tweet Sarcasm. We emphasize the modularity of our solution as one of the advantages of our approach. New functionality could be easily added to the current configuration, tracking new aspects of polarity detection that was left unattended in the current state of development.

Despite the overall goodness of the system, there is a generalized degradation in the evaluation results between 2014 and 2015 Twitter datasets. This result is very interesting and encouraging for future lines of work, as there exists a clear need in research of new models which provide better abstraction of the data and improve the adaptation to new contexts that differ substantially the training vectors.

References

Awais Athar. 2011. Sentiment Analysis of Citations using Sentence Structure-Based Features. In *Proceedings of the ACL 2011 student session*, pages 81–87.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *LREC*, volume 10, pages 2200–2204.

Yejin Choi and Claire Cardie. 2008. Learning with Compositional Semantics as Structural Inference for Subsentential Sentiment Analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 793–801.

Isaac G Councill, Ryan McDonald, and Leonid Velikovich. 2010. What's Great and What's Not: Learning to Classify the Scope of Negation for Improved Sentiment Analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 51–59.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter Sentiment Classification using Distant Supervision. *CS224N Project Report, Stanford*, pages 1–12.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for Segmenting and Labeling Sequence Data.

Emanuele Lapponi, Jonathon Read, and Lilja Ovrelid. 2012a. Representing and Resolving Negation for Sentiment Analysis. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on*, pages 687–692.

Emanuele Lapponi, Erik Velldal, Lilja Øvrelid, and Jonathon Read. 2012b. Uio 2: Sequence-Labeling Negation Using Dependency Features. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 319–327.

Tetsuji Nakagawa, Kentaro Inui, and Sadao Kurohashi. 2010. Dependency Tree-Based Sentiment Classification using crfs with Hidden Variables. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 786–794.

Lluís Padró and Evgeny Stanilovsky. 2012. Freeling 3.0: Towards Wider Multilinguality. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2012)*, Istanbul, Turkey, May.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: Sentiment Classification using Machine Learning Techniques. In *Proceedings of the*

ACL-02 Conference on Empirical methods in Natural Language Processing-Volume 10, pages 79–86.

Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif M Mohammad, Alan Ritter, and Veselin Stoyanov. 2015. Semeval-2015 Task 10: Sentiment Analysis in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, SemEval '2015, Denver, Colorado, June.

Grigori Sidorov, Sabino Miranda-Jiménez, Francisco Viveros-Jiménez, Alexander Gelbukh, Noé Castro-Sánchez, Francisco Velásquez, Ismael Díaz-Rangel, Sergio Suárez-Guerra, Alejandro Treviño, and Juan Gordon. 2013. Empirical Study of Machine Learning Based Approach for Opinion Mining in Tweets. In *Advances in Artificial Intelligence*, pages 1–14.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality over a Sentiment Treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642.

Charles Sutton and Andrew McCallum. 2011. An Introduction to Conditional Random Fields. *Machine Learning*, 4(4):267–373.

Duyu Tang, Furu Wei, Bing Qin, Ting Liu, and Ming Zhou. 2014. Coooolll: A Deep Learning System for Twitter Sentiment Classification. *SemEval 2014*, page 208.

Xiaodan Zhu, Svetlana Kiritchenko, and Saif M Mohammad. 2014. Nrc-canada-2014: Recent Improvements in the Sentiment Analysis of Tweets. *SemEval 2014*, page 443.