LT3: Sentiment Analysis of Figurative Tweets: piece of cake #NotReally

Cynthia Van Hee, Els Lefever and Véronique hoste
LT^3, Language and Translation Technology Team
Department of Translation, Interpreting and Communication – Ghent University
Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
Firstname.Lastname@UGent.be

Abstract

This paper describes our contribution to the SemEval-2015 Task 11 on sentiment analysis of figurative language in Twitter. We considered two approaches, classification and regression, to provide fine-grained sentiment scores for a set of tweets that are rich in sarcasm, irony and metaphor. To this end, we combined a variety of standard lexical and syntactic features with specific features for capturing figurative content. All experiments were done using supervised learning with LIBSVM. For both runs, our system ranked fourth among fifteen submissions.

1 Introduction

Handling figurative language is currently one of the most challenging tasks in NLP. Figurative language is often characterized by linguistic devices such as sarcasm, irony, metaphors, and humour. Their meaning goes beyond the literal meaning and is therefore often hard to capture, even for humans. However, as an increasing part of our daily communication takes place on social media (e.g. Twitter, Facebook), which are prone to figurative language use, there is an urgent need for automatic systems that recognize and understand figurative online content. This is especially the case in the field of sentiment analysis where the presence of figurative language in subjective text can significantly undermine the classification accuracy.

Understanding figurative language often requires world knowledge, which cannot easily be accessed by machines. Moreover, figurative language rapidly evolves due to changes in vocabulary and language, which makes it difficult to train machine learning algorithms. Nevertheless, the identification of non-literal uses of language has attracted a fair amount of research interest recently. Veale (2012) investigated the relation between irony and our stereotypical knowledge of a domain and showed how the insight in stereotypical norms helps to recognize and understand ironic utterances. Reyes et al. (2013) built an irony model for Twitter for which they relied on a set of textual features for capturing ironic tweets. Their model obtained promising results concerning recall (84%). In what relates to the detection of metaphors, Turney et al. (2011) introduced an algorithm for distinguishing between metaphorical and literal word usages based on the degree of abstractness of a word's context. More recent work by Tsvetkov et al. (2014) presents a cross-lingual model based on lexical semantic word features for metaphor detection in English, Spanish, Farsi and Russian.

To date, most studies on figurative language use have focussed on the detection of linguistic devices such as sarcasm, irony and metaphor. By contrast, only a few studies have investigated how these devices affect sentiment analysis. Indeed, as stated by Maynard (2014), it is not sufficient to determine whether a text contains sarcasm or not. Instead, we need to measure its impact on sentiment analysis if we want to improve the state-of-the-art in sentiment analysis systems.

In this paper we describe our contribution to the SemEval-2015 shared task: *Sentiment Analysis of Figurative Language in Twitter* (Ghosh et al., 2015).

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 684–688,
Denver, Colorado, June 4-5, 2015. ©2015 Association for Computational Linguistics

Our objective is to provide fine-grained sentiment scores for a set of tweets that are rich in sarcasm, irony and metaphor. The datasets for training, development and testing were provided by the task organizers. The training dataset contains 8,000 tweets (5,000 sarcastic, 1,000 ironic and 2,000 metaphorical) labeled with a sentiment score between -5 and 5. This training set was provided with both integer and real-valued sentiment scores. The trial and test sets were comparable to the training corpus and contain 1,000[1] and 4,000 labeled instances, respectively. All experiments were done using LIBSVM (Chang and Lin, 2011).

We submitted two runs for the competition. To this end, we built two models based on supervised learning: 1) a classification-based (C-SVC) and 2) a regression-based approach (epsilon-SVR). For both models, we implemented a number of word-based, lexical, sentiment and syntactic features in combination with specific features for capturing figurative content such as sarcasm. Evaluation was done by calculating the cosine similarity distance between the predicted and the gold-standard sentiment labels.

The remainder of this paper is structured as follows: Section 2 presents our system description whereas Section 2.2 gives an overview of the features we implemented. The experimental setup is described in Section 3, followed by our results in Section 4. Finally, we draw conclusions in Section 5 where we also suggest some directions for future research.

2 System Description

The main purpose of this paper was to develop a system for the fine-grained sentiment classification of figurative tweets. We tackled this problem by using classification and regression approaches and provided each instance with a sentiment score between -5 and 5. In addition to more standard NLP features (bags-of-words, PoS-tags, etc.), we implemented a number of features for capturing the figurative character of the tweets. In this section, we outline our sentiment analysis pipeline and describe the linguistic preprocessing and feature extraction.

2.1 Linguistic Preprocessing

All tweets were tokenized and PoS-tagged using the Carnegie Mellon University Twitter Part-of-Speech-Tagger (Gimpel et al., 2011). Lemmatization was done using the LT3 LeTs Preprocess Toolkit (Van de Kauter et al., 2013). We used a caseless parsing model of the Stanford parser (de Marneffe et al., 2006) for a dependency representation of the messages. As a final step, we tagged all named entities using the Twitter NLP tools for Named Entity Recognition (Ritter et al., 2011).

2.2 Features

As a first step, we implemented a set of features that have shown to perform well for sentiment classification in previous research (Van Hee et al., 2014). These include word-based features (e.g. bag-of-words), lexical features (e.g. character flooding), sentiment features (e.g. an overall sentiment score per tweet, based on existing sentiment lexicons), and syntactic features (e.g. dependency relation features)[2]. To provide some abstraction, we also added PoS n-gram features to the set of bag-of-words features.

Nevertheless, as a substantial part of the data we are confronted with is of a figurative nature, we implemented a series of additional features for capturing potential clues, for example of sarcasm, in the tweets[3].

Contrast – Binary feature indicating whether a contrastive sentiment (i.e. at least one positive and one negative sentiment word) is contained by the instance.

Interjection Count – Numeric feature indicating how many interjections are contained by an instance. This value is normalized by dividing it by the number of tokens in the instance. As stated by (Carvalho et al., 2009), interjections may be potential clues for irony detection.

Sarcasm Hashtag – Binary feature indicating whether an instance contains a hashtag that may indicate the presence of sarcasm. To this end, a list of

[1] As some tweets were made inaccessible by their creators, we were able to download only 914 of them

[2] For a detailed description of these features we refer to Van Hee et al. (2014).

[3] A number of these features (i.e. *contradiction*, *sudden change*, and *temporal imbalance*) are inspired by Reyes et al. (2013).

≈ 100 sarcasm-related hashtags was extracted from the training data.

Punctuation Mark Count – Normalized numeric feature indicating the number of punctuation marks that are contained by an instance.

Emoticon count – Normalized numeric feature indicating the number of emoticons that are contained by an instance.

Contradiction – Binary feature that indicates whether an instance contains a linguistic contradiction marker (i.e. words like *nonetheless*, *yet*, *however*).

Sudden Change – Binary feature that indicates whether an instance contains a linguistic marker of a sudden change in the narrative of the tweet (i.e. words like *suddenly*, *out of the blue*).

Temporal Imbalance – Binary feature indicating the presence of a temporal imbalance (i.e. both present and past tenses are used) in the narrative of a message.

Polysemy – Normalized numeric feature indicating how many polyseme words are contained by an instance. As polyseme are considered those words that have more than seven different meanings according to WordNet[4], which may be an indication of metaphorical language.

3 Experimental Setup

As the training instances were provided with both integer and real-valued sentiment scores, we used two different approaches to the fine-grained sentiment labeling. Firstly, we implemented a classification approach where each tweet had to be given a sentiment label on an eleven-point scale ranging from -5 to 5. Secondly, we used regression to predict a real-valued sentiment score for each tweet, which could be any numeric value between -5 and 5.

Two feature sets were used throughout the experiments: firstly, we included a number of word-based, lexical, sentiment and syntactic features (we refer to these as the *sentiment* feature set). Secondly, we implemented an additional set of features for capturing possibly figurative content such as irony and metaphors. These features are referred to as the *figurative* feature set.

Using 5-fold cross-validation on the training data, we performed a grid search to find the optimal cost and gamma parameters for both classification (c = 0.03, g = 0.008) and regression (c = 8, g = 0.063). For regression, an optimal epsilon value of p = 0.5 was determined.

As a first approach to evaluating our features, we used a subset of the trial data[5]. Secondly, we (randomly) split the data into 90% for training and 10% for testing. We calculated a baseline using the majority class label -3 (see Table 1). Tables 2 and 3 present the results on the training and trial data that were obtained throughout the experiments both for classification and for regression.

Evaluation Set	Cosine Similarity
Trial data	0.59
10% training set	0.80
Averaged baseline	**0.70**

Table 1: Majority class baseline.

Evaluation Set	feature set	Cosine Similarity
Trial data	sentiment	0.72
	figurative	0.74
10% training set	sentiment	0.82
	figurative	0.83

Table 2: Experimental results for classification (after a parameter grid search).

Evaluation Set	feature set	Cosine Similarity
Trial data	sentiment	0.75
	figurative	0.74
10% training set	sentiment	0.85
	figurative	0.84

Table 3: Experimental results for regression (after a parameter grid search).

As the table shows, adding figurative language specific features proves to be beneficial for classification. For regression, by contrast, adding more features does not improve the results on the training and trial data. However, both approaches clearly outperform the baseline.

[4] Fellbaum, C. (1998)

[5] We only considered the tweets that were not included by the training data.

4 Competition Results

We submitted two runs for this task. For our first run, we implemented a classification approach whereas we used regression for the second run. As the official test data also contains a substantial part of regular Twitter data, we included both the standard sentiment feature set and the figurative feature set.

Our competition results can be found in Tables 4 and 5.

	Overall	Sarcasm	Irony	Metaphor	Other
Cosine Similarity	**0.66** (4/15)	0.89	0.90	0.44	0.35
MSE	**3.40** (4/15)	1.29	1.22	5.67	5.44

Table 4: Competition results for classification.

	Overall	Sarcasm	Irony	Metaphor	Other
Cosine Similarity	**0.65** (4/15)	0.87	0.86	0.36	0.36
MSE	**2.91** (4/15)	1.29	1.08	4.79	4.50

Table 5: Competition results for regression.

As shown in tables 4 and 5, our system achieved an overall cosine similarity score of 0.66 and 0.65 for the classification-based and regression-based approaches respectively and ranked fourth among fifteen submissions for both runs. When considering the competition results per category, we see that our system performs particularly well on the sarcasm and irony classes. For the latter, our classification performance (cosine similarity = 0.90) corresponds with that of the best reported system.

5 Conclusions and Future Work

We experimented with two experimental setups to compare the performance of a sentiment classifier using 1) more standard sentiment features and 2) features that may capture sarcastic content. The results of our experiments show that adding features that are specific to figurative language improves the performance of our classification approach. However, it does not improve the performance for regression.

An error analysis revealed that our system's performance benefits from the information provided by sentiment lexicon features. Given the high distribution of the negative class labels in this corpus, some positive instances are incorrectly assigned a negative class label:

- *Im not about that life though lol, Im literally a natural woman and I am proud of it :)* **(-3)**

Another remark that should be made is that some of our irony-specific features are possibly too coarse-grained. The contrast feature for instance, was sometimes activated even though the tweet under investigation was meant rather literally than sarcastically:

- *RT @laurenwalter: underwater walking was* ***pretty bloody amazing****! literally wanted to stay under there! was such an experience!! loved it!!*

The contrast feature was activated for this tweet since *bloody* was identified as a negative sentiment word whereas *pretty* and *amazing* are positive sentiment words. This problem may be solved by only considering the head of the adjectival phrase (*amazing*) as a sentiment word.

In this paper, we developed a sentiment analysis pipeline that takes irony and sarcasm clues into account to provide a fine-grained sentiment score for tweets. In future research, it would be interesting to implement a cascaded approach where 1) the output of a sarcasm detection system is used as a feature for a sentiment classifier or 2) a sarcasm detection system is used as a post-processing step where the sentiment label given by a regular sentiment classifier is flipped if the utterance is meant sarcastically.

Moreover, we will search for better features for modeling sarcasm in tweets and we aim to rebalance the data to approximate a realistic distribution of sarcastic messages in a random stream of Twitter messages.

To improve sentiment classification of metaphorical tweets, a classifier might benefit from word sense disambiguation and knowledge about stereotypes and commonly used similes.

Finally, we aim to perform feature selection since abounding bag-of-words features often suffer from overfitting. This way, they may introduce noise and hence decrease the classification accuracy.

References

Paula Carvalho, Luís Sarmento, Mário J. Silva, and Eugénio de Oliveira. 2009. Clues for detecting irony in user-generated contents: Oh...!! it's "so easy" ;-). In *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*, TSA '09, pages 53–56, New York, NY, USA.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating Typed Dependency Parses from Phrase Structure Parses. In *Proc. of LREC'06*, pages 449–454, Genoa, Italy.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*.

A. Ghosh, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, Antonio Reyes, and John Barnden. 2015. Semeval-2015 task 11: Sentiment analysis of figurative language in Twitter. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval-2015)*, Denver, Colorado, USA.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for Twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pages 42–47, Stroudsburg, PA, USA.

Diana Maynard and Mark Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in Twitter. *Language Resources and Evaluation*, pages 238–269.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. *ACL 2014*, pages 248–258.

Peter D. Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. 2011. Literal and metaphorical sense identification through concrete and abstract context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 680–690, Stroudsburg, PA, USA.

Marjan Van de Kauter, Geert Coorman, Els Lefever, Bart Desmet, Lieve Macken, and Véronique Hoste. 2013. LeTs Preprocess: the multilingual LT3 linguistic preprocessing toolkit. *Computational Linguistics in the Netherlands Journal*, 3:103–120.

Cynthia Van Hee, Marjan Van de Kauter, Orphée De Clercq, Els Lefever, and Véronique Hoste. 2014. LT3: Sentiment classification in user-generated content using a rich feature set. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 406–410, Dublin, Ireland.

Tony Veale. 2012. Detecting and generating ironic comparisons: An application of creative information retrieval. In *AAAI Fall Symposium: Artificial Intelligence of Humor*, volume FS-12-02 of *AAAI Technical Report*.