LLT-PolyU: Identifying Sentiment Intensity in Ironic Tweets

Hongzhi Xu, Enrico Santus, Anna Laszlo and Chu-Ren Huang
The Department of Chinese and Bilingual Studies
The Hong Kong Polytechnic University
{hongz.xu, esantus}@gmail.com
mandarin1985@yahoo.de
churenhuang@gmail.com

Abstract

In this paper, we describe the system we built for Task 11 of SemEval2015, which aims at identifying the sentiment intensity of figurative language in tweets. We use various features, including those specially concerned with the identification of irony and sarcasm. The features are evaluated through a decision tree regression model and a support vector regression model. The experiment result of the five-cross validation on the training data shows that the tree regression model outperforms the support vector regression model. The former is therefore used for the final evaluation of the task. The results show that our model performs especially well in predicting the sentiment intensity of tweets involving irony and sarcasm.

1 Introduction

Sentiment analysis aims to identify the polarity and intensity of certain texts in order to shed light on people's sentiments, perceptions, opinions, and beliefs about a particular product, service, scheme, etc. Knowing what people think can, in fact, help companies, political parties, and other public entities in strategizing and decision making.

While impressive results have been achieved in analysing literal texts (Abbasi et al., 2008; Yan et al., 2014), the study of polarity shifting in sentiment analysis still requires much research. For example, Li, et.al. (2010), explores the polarity shifters in English which significantly improve the performance of sentiment analysis. Besides, figurative uses of language, such as irony or sarcasm, are also able to invert the polarity of the surface text. Theoretical research in irony and sarcasm often emphasize that humans have difficulties in deciphering messages with underlying meaning (Hay, 2001; Kotthoff, 2003; Kreuz and Caucci, 2007). Factors that can facilitate the understanding of these messages include prosody (e.g. stress or intonation), kinesics (e.g. facial gestures), co-text (i.e. immediate textual environment) and context (i.e. wider environment), as well as cultural background. Computers, however, cannot always rely on this kind of information.

Currently, there is no method that can guarantee the unequivocal recognition of irony or sarcasm. Training a computer to perform such a highly pragmatic task does indeed pose a challenge to computational linguists. A good number of studies have been recently devoted to finding a solution to the problem. Most of them have focused on tweets (González-Ibáñez et al., 2011; Reyes et al., 2013; Liebrecht et al., 2013; Riloff et al., 2013; Barbieri et al., 2014; Vanzo et al., 2014).

Identifying figurative language in short messages (generally consisting of no more than 140 characters) that do not make use of conventional language, but employ "little space-consuming" elements, such as emoticons (":D"), abbreviations ("abbr.") and slang ("slng") is not a self-evident task. The reason why none of these studies has proved to be the representative method that could widely be adopted and applied by other researchers is that they have not yet reached optimal results. Thus, the devising of a computational model able to accurately detect polarity is very much on-going.

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 673–678,
Denver, Colorado, June 4-5, 2015.

This paper describes the model we developed for Task 11 of SemEval-2015 (Ghosh et al., 2015), which is concerned with the Sentiment Analysis of Figurative Language in Twitter. Our model came first in the SemEval-2015 task for irony and third in the overall ranking, showing that the features we proposed produce more reliable results in sentiment analysis of ironic tweets.

2 Related Work

Irony is defined by Quintilian in the first century CE as "saying the opposite of what you mean" (Quintilian, 1922). It violates the expectations of the listener by flouting the maxim of quality (Grice, 1975; Stringfellow Jr, 1994; Gibbs and Colston, 2007; Tungthamthiti et al., 2014). In the same fashion, sarcasm is generally understood as the use of irony "to mock or convey contempt" (Stevenson, 2010).

While irony and sarcasm are well studied in linguistics and psychology, their automatic identification through Natural Language Processing methods is a relatively novel task (Pang and Lee, 2008). Not to mention that irony and sarcasm pose a difficult problem in Sentiment Analysis of micro blogging and social media (Barbieri et al., 2014).

Up to this date, several approaches have been proposed to automatically identify irony and sarcasm in tweets and comments. Carvalho et al. (2009), for example, proposed to identify irony in comments to newspaper articles by relying on the presence of emoticons, onomatopoeic expressions, and heavy punctuation in the text surface. Hao and Veale (2010) have investigated similes of the form "x as y" in a large corpus, proposing a method to automatically discriminate ironic from non-ironic similes. Tsur et al. (2010) proposed a semi-supervised approach for the automatic recognition of sarcasm in Amazon product reviews, exploiting some features that were specific to Amazon. Their method employed two modules: a semi-supervised acquisition of sarcastic patterns and a classifier. This method was then applied to tweets by Davidov et al. (2010), achieving even better results. González-Ibáñez et al. (2011) constructed a corpus of sarcastic tweets and used it to compare judgements made by humans and machine learning algorithms, concluding that none of them performed well.

More recently, Reyes et al. (2013) defined a complex model for identifying sarcasm which goes far behind the surface of the text and takes into account features on four levels: signatures, degree of unexpectedness, style, and emotional scenarios. They have demonstrated that these features do not help the identification in isolation. However, they do if they are combined in a complex framework. Barbieri and Saggion (2014) focused their approach on the use of lexical and semantic features, such as the frequency of the words in different reference corpora, the length of the words, and the number of related synsets in WordNet (Miller and Fellbaum, 1998).

Finally, Buschmeier et al. (2014) assessed the impact of features used in previous studies, and they provide an important baseline for irony detection in English.

Many datasets for the study of irony and sarcasm in Twitter are nowadays available. Thanks to the use of hashtags, it is easier to collect data with specific characteristics in Twitter. Reyes et al. (2013), for example, created a corpus of 40.000 tweets with four categories: Irony, Education, Humour, and Politics. Among the other resources, it is worth mentioning the sarcastic Amazon product reviews collected by Filatova (2012) and the Italian examples collected and annotated by Gianti et al. (2012), later used in Bosco et al. (2013).

3 Methodology

3.1 Data Pre-processing

Considering the unregulated and arbitrary nature of the texts we are working with, we use some heuristic rules to pre-process them. These rules help us get more reliable syntactic structures when calling the syntactic parser.

Twitter users often use repeated vowels (e.g. "*loooove*") or capitalization (e.g. "*LOVE*") to emphasize certain sentiments or emotions. The normalization consists of removing the repeated vowels (e.g. from "*loooove*" to "*love*") and the capitalization (e.g. from "*LOVE*" to "*love*"). The normalized forms can help improve the parsing accuracy. Moreover, they are saved in a special feature bag as they are important indicators of sentiments, especially when they are in sentiment lexicons. Other special uses of language in tweets include the socalled heavy punctuation and emoticons. In our system, we substitute every combination of exclamation and question marks (e.g. "*?!?!!*") with the form "*?!*". We also compiled an emoticon dictionary based on training data and internet resources.

Another step that we considered relevant at this point is the maximal matching segmentation. The segmentation is, in fact, often lost in tweets, as white spaces and punctuation are not always used in their customary format (e.g. "*yeahright*"). In order to get rid of this problem, we tried to segment all the out of vocabulary tokens through a maximal matching algorithm according to an English dictionary (e.g. the token "*yeahright*" would be segmented as "*yeah right*").

Finally, we use Stanford parser (Klein and Manning, 2003) to get the POS tags and dependency structures of the normalized tweets.

3.2 Feature Set

After the pre-processing, we then extract features of the following kinds.

UniToken Token uni-grams are the basic features in our approach. The normalized forms of the emphasized tokens are put in a special bag with tags describing their emphasis types {duplicate_vowel, capitalized, heavy_punctuation, emoticon}

BiToken Bi-grams of the normalized tokens are also used as features.

DepTokenPair The "parent-child" pairs based on dependency structures are also used as features.

PolarityWin In order to identify the polarity values of tokens, we used four sentiment dictionaries: Opinion Lexicon (Hu and Liu, 2004), Afinn (Nielsen, 2011), MPQA (Wiebe et al., 2005), and SentiWordnet (Baccianella et al., 2010). Their union and their intersection are also used as two additional dictionaries. A window size of five is used to verify whether negations are present. If a negation is present the resulting value is set to zero. Six features are used to save the sum polarity values of a tweet based on the six dictionaries respectively. Besides, we also use features recording the polarity contribution of different POS tags. For example, one possible feature-value pair can be (adj-$mpqa$, 1.0) meaning that according to the dictionary $MPQA$, the sum of the polarity contributed by adjectives in the current tweet is 1.0.

PolarityDep This feature set is similar to $PolarityWin$, but it differs in that the negation is checked in the dependency structure.

PolarShiftWin This feature set is designed for irony which has been discussed in (Riloff et al., 2013). Let us consider the tweets (1) "I love working for eight hours without any break" and (2) "I hate people giving me such a big surprise". In these tweets the verbs "love" (positive) and "hate" (negative) are used with reference to a negative and a positive clause ("working for eight hours without any break" and "people giving me such a big surprise") respectively. Based on a 5-window we check whether a shift of polarity is present.

PolarShiftDep This feature set is similar to $PolarShiftWin$, but it differs in that the shift is checked in the dependency structure.

3.3 Feature Normalization and Evaluation

In order to avoid noise and sparseness, only features that occur at least 3 times are kept. All the feature values are normalized into the range [-1, 1] according to the formula shown in Equation 1, where $f_{i,j}$ is the value of feature j in the ith example, and N is the sample size.

$$norm(f_{i,j}) = \frac{f_{i,j}}{\max\limits_{1 \le k \le N} |f_{k,j}|} \tag{1}$$

We use the correlative coefficient (Pearson's r) measure to rank all the features. Then, we can use the threshold value of r to rule out less important features. The calculation of r is described in Equation 2, where X and Y are the two variables that are evaluated, X_i is the ith sample value of X, Y_i is the ith sample value of Y and N is the sample size.

$$r(X,Y) = \frac{\sum_{i=1}^{N}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{N}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{N}(Y_i - \bar{Y})^2} \tag{2}$$

The goal of the first experiment is to find the optimal threshold value of r with all the features as listed in 3.2. Two different models are used: Decision Tree

Feature Set	Features	mse	cosine
Baseline	N/A	1.9847	0.8184
UniToken	136	1.6821	0.8507
+BiToken	410	1.7007	0.8485
+DepTokenPair	409	1.6733	0.8514
+PolarityWin	582	1.6573	0.8524
+PolarityDep	748	1.6436	0.8536
+PolarShiftWin	825	1.6403	0.8542
+PolarShiftDep	841	1.6393	0.8543

Table 1: Experiment result of the 5-fold cross validation by RegTree and SVR on the training data.

Regression model (RepTree) implemented in Weka (Hall et al., 2009) and Support Vector Regression model (SVR) implemented in LibSVM (Chang and Lin, 2011). The result is shown in Figure 1. The best performance is obtained with the value of r between 0.03 and 0.04 with the RepTree model. The experiment also shows that RepTree always outperforms SVR (i.e. higher $cosine$ value and lower $rmse$ value). Therefore, in the following experiments and in the evaluation the RepTree model is adopted.

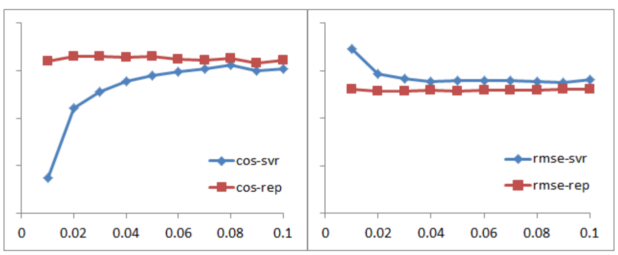

Figure 1: Effect of Pearson value threshold on the overall performance in cosine (left) and root mean squared error (right).

In the second experiment, we use $r = 0.035$ as threshold for feature selection by testing how different kinds of features contribute to the overall performance. The features listed in Section 3.2 are gradually added and their contribution is assessed. If the new feature does not improve the performance, it is removed in the next running. The results of the second experiment are shown in Table 1. The baseline is obtained with a naive prediction using the average polarity value of the training data. As can be seen, only $BiToken$ harms the performance, while all other features contribute to its improvement.

category	mse	cosine
Sarcasm	0.997	0.896
Irony	0.671	0.918
Metaphor	3.917	0.535
Other	4.617	0.290
Overall	2.602	0.687

Table 2: Test result of SemEval Task 11.

3.4 Evaluation Result

Based on the described analysis, for the final test we used RepTree and all the feature sets, except for $BiToken$. The threshold for feature frequency is set to 3 and the r value for feature selection is set to 0.035. Finally, the trained model on the 8,000 tweets is used to predict the sentiment intensity of the evaluation dataset which includes 4,000 tweets. The results are shown in Table 2. Among the fifteen participants in the SemEval task on *Sentiment Analysis of Figurative Language in Twitter*, our model achieves the best performance in the identification of irony, and ranks third in the overall performance.

4 Conclusions

In this paper, we introduced our model for the *Sentiment Analysis of Figurative Language in Twitter* following the track of Task 11 of SemEval 2015. We first used heuristic rules to pre-process the tweets by identifying and normalizing the emphasized tokens. Then, features were extracted based on both window and dependency structures. We adopted polarity shift features with special consideration on the identification of irony. As expected, our system performed best in predicting the sentiment intensity of tweets containing irony according to the evaluation. This confirms the robustness of our design and points to promising development of automatic processing of irony in the future.

Acknowledgments

The work is supported by a General Research Fund (GRF) sponsored by the Research Grants Council (Project no. 543512 & 543810). This work is partially supported by HK PhD Fellowship Scheme, under PF11-00122 and PF12-13656.

References

Ahmed Abbasi, Hsinchun Chen, and Arab Salem. 2008. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Transactions on Information Systems (TOIS)*, 26(3):12:1–12:34.

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.

Francesco Barbieri and Horacio Saggion. 2014. Modelling irony in twitter. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 56–64.

Francesco Barbieri, Francesco Ronzano, and Horacio Saggion. 2014. Italian irony detection in twitter: a first approach. In *The First Italian Conference on Computational Linguistics CLiC-it 2014 & the Fourth International Workshop EVALITA 2014*, pages 28–32.

Cristina Bosco, Viviana Patti, and Andrea Bolioli. 2013. Developing corpora for sentiment analysis and opinion mining: the case of irony and senti-tut. *IEEE Intelligent Systems*, 28(2):55–63.

Konstantin Buschmeier, Philipp Cimiano, and Roman Klinger. 2014. An impact analysis of features in a classification approach to irony detection in product reviews.

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*, pages 53–56. ACM.

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

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 107–116.

Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Proceedings of Language Resources and Evaluation Conference*, pages 392–398.

Aniruddha 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), Co-located with NAACL and *SEM*, Denver, Colorado, US, June 4-5.

Andrea Gianti, Cristina Bosco, Viviana Patti, Andrea Bolioli, and Luigi Di Caro. 2012. Annotating irony in a novel italian corpus for sentiment analysis. In *Proceedings of the 4th Workshop on Corpora for Research on Emotion Sentiment and Social Signals*, pages 1–7.

Raymond W Gibbs and Herbert L Colston. 2007. *Irony in language and thought: A cognitive science reader*. Psychology Press.

Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers*, volume 2, pages 581–586.

H Paul Grice, 1975. *Logic and conversation*, volume 3 of *Syntax and Semantics*, pages 41–58. Academic Press, New York.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The weka data mining software: An update. *SIGKDD Explorations*, 11.

Yanfen Hao and Tony Veale. 2010. An ironic fist in a velvet glove: Creative mis-representation in the construction of ironic similes. *Minds and Machines*, 20(4):635–650.

Jennifer Hay. 2001. The pragmatics of humor support. *International Journal of Humor Research*, 14:1–27.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430.

Helga Kotthoff. 2003. Responding to irony in different contexts: On cognition in conversation. *Journal of pragmatics*, 35(9):1387–1411.

Roger J Kreuz and Gina M Caucci. 2007. Lexical influences on the perception of sarcasm. In *Proceedings of the Workshop on computational approaches to Figurative Language*, pages 1–4.

Shoushan Li, Sophia Yat Mei Lee, Ying Chen, Chu-Ren Huang, and Guodong Zhou. 2010. Sentiment classification and polarity shifting. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 635–643.

Christine Liebrecht, Florian Kunneman, and Antal van den Bosch. 2013. The perfect solution for detecting sarcasm in tweets# not. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. New Brunswick, NJ: ACL.

George Miller and Christiane Fellbaum. 1998. Wordnet: An electronic lexical database.

Finn Årup Nielsen. 2011. A new anew: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages*.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Quintilian. 1922. *With An English Translation. Harold Edgeworth Butler*. Cambridge, Mass., Harvard University Press; London, William Heinemann, Ltd.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation*, 47(1):239–268.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of Conference on Emperical Methods for Natural Language Processing (EMNLP)*, pages 704–714.

Angus Stevenson. 2010. *Oxford dictionary of English*. Oxford University Press.

Frank Stringfellow Jr. 1994. *The meaning of irony: A psychoanalytic investigation*. SUNY Press.

Oren Tsur, Dmitry Davidov, and Ari Rappoport. 2010. Icwsm-a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In *In International AAAI Conference on Web and Social Media*.

Piyoros Tungthamthiti, Shirai Kiyoaki, and Masnizah Mohd. 2014. Recognition of sarcasms in tweets based on concept level sentiment analysis and supervised learning approaches. In *Proceedings of Pacific Asia Conference on Language, Information and Computing*, Phuket, Thailand.

Andrea Vanzo, Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2014. A context based model for sentiment analysis in twitter for the italian language. In R. Basili, A. Lenci, and B. Magnini, editors, *Proceedings of the First Italian Conference on Computational Linguistics CLiC-it & the Fourth International Workshop EVALITA*, pages 379–383, Pisa. Pisa University Press.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210.

Gongjun Yan, Wu He, Jiancheng Shen, and Chuanyi Tang. 2014. A bilingual approach for conducting chinese and english social media sentiment analysis. *Computer Networks*, 75:491–503.