10.12871/CLICIT201416

Italian Irony Detection in Twitter: a First Approach*

Francesco Barbieri, Francesco Ronzano, Horacio Saggion
Universitat Pompeu Fabra, Barcelona, Spain
name.surname@upf.edu

Abstract

English. Irony is a linguistic device used to say something but meaning something else. The distinctive trait of ironic utterances is the opposition of literal and intended meaning. This characteristic makes the automatic recognition of irony a challenging task for current systems. In this paper we present and evaluate the first automated system targeted to detect irony in Italian Tweets, introducing and exploiting a set of linguistic features useful for this task.

Italian. *L'ironia è una figura retorica mediante la quale si vuole conferire a una espressione un significato differente da quello letterale. Il riconoscimento automatico dell'ironia è reso difficile dalla sua principale caratteristica: il contrasto tra significato inteso e significato letterale. In questo studio proponiamo e valutiamo il primo sistema per il riconoscimento automatico di Tweets ironici in italiano.*

1 Introduction

Sentiment Analysis is the interpretation of attitudes and opinions of subjects on certain topics. With the growth of social networks, Sentiment Analysis has become fundamental for customer reviews, opinion mining, and natural language user interfaces (Yasavur et al., 2014). During the last decade the number of investigations dealing with sentiment analysis has considerably increased, targeting most of the time English language. Comparatively and to the best of our knowledge there are only few works for the Italian language.

*The research described in this paper is partially funded by the Spanish fellowship RYC-2009-04291, the SKATER-TALN_UPF project (TIN2012-38584-C06-03), and the EU project Dr. Inventor (n. 611383).

In this paper we explore an important sentiment analysis problem: *irony detection*. Irony is a linguistic device used to say something when meaning something else (Quintilien and Butler, 1953). Dealing with figurative languages is one of the biggest challanges to correctly determine the polarity of a text: analysing phrases where literal and indented meaning are not the same, is hard for humans, hence even harder for machines. Moreover, systems able of detect irony can benefit also other A.I. areas like Human Computer Interaction.

Approaches to detect irony have been already proposed for English, Portuguese and Dutch texts (see Section 2). Some of these systems used words, or word-patterns as irony detection features (Davidov et al., 2010; González-Ibáñez et al., 2011; Reyes et al., 2013; Buschmeier et al., 2014). Other approaches, like Barbieri and Saggion (2014a), exploited lexical and semantic features of single words like their frequency in reference corpora or the number of associated synsets. Relying on the latter method, in this paper we present the first system for automatic detection of irony in Italian Tweets. In particular, we investigate the effectiveness of Decision Trees in classifying Tweets as ironic or not ironic, showing that classification performances increase by considering lexical and semantic features of single words instead of pure bag-of-words (BOW) approaches. To train our system, we exploited as ironic examples the Tweets from the account of a famous collective blog named Spinoza and as not ironic examples the Tweets retrieved from the timelines of seven popular Italian newspapers.

2 Related Work

The standard definition of irony is "saying the opposite of what you mean" (Quintilien and Butler, 1953). Grice (1975) believes that irony is a rhetorical figure that violates the maxim of quality, while Giora (1995) says that irony can be any

form of negation with no negation markers. Wilson and Sperber (2002) defined irony as echoic utterance that shows a negative aspect of someone's else opinion. Utsumi (2000) and Veale and Hao (2010a) stated that irony is a form of pretence that is violated.

Irony has been approached computationally by Veale and Hao (2010b) who proposed an algorithm for separating ironic from non-ironic similes in English, detecting common terms used in this ironic comparison. Reyes et al. (2013) proposed a model to detect irony in English Tweets, pointing out that skipgrams which capture word sequences that contain (or skip over) arbitrary gaps, are the most informative features. Barbieri and Saggion (2014a) and Barbieri and Saggion (2014b) designed a model that avoided the use of the words (or pattern of words) as the use of single words or word-patterns as features. They focused on the lexical and semantic information that characterises each word in an Tweet, like its frequency in different corpora, its length, the number of associated synsets, etc. The system of Buschmeier et al. (2014) included features proposed in previous systems and gave for the first time a baseline for the irony detection problem in English (best F1-measure obtained was 0.74). Little research has been carried out on irony detection in languages other than English. Carvalho et al. (2009) and de Freitas et al. (2014) dealt with irony in Portuguese newspapers. Liebrecht et al. (2013) designed a model to detect irony in Dutch Tweets.

Gianti et al. (2012) collected and annotate a set of ironic examples from a common collective Italian blog. This corpus is also used in Bosco et al. (2013) for the study of sentiment analysis and opinion mining in Italian.

3 Data and Text Processing

The corpus[1] we used is composed of 25,450 Tweets: 12.5% are ironic and 87.5% non-ironic. The set of ironic examples (3,185) is an aggregation of the posts from the Twitter accounts "spinozait" and "LiveSpinoza". Spinoza is an Italian collective blog that includes posts of sharp satire on politics (the posts are suggested by the community and a group of volunteers filter the content to be published). Spinoza is a very popular blog and there is a collective agreement on the irony of its posts (Bosco et al., 2013). The non-ironic examples (22,295) are Tweets retrieved from Twitter accounts of the seven most popular Italian daily newspapers, including "Corriere della Sera", "Gazzetta dello Sport", "Il Messaggero", "Repubblica", "Il Resto del Carlino", "Il Sole 24 Ore", and "La Stampa". Almost the totality of these posts do not contain irony, they only describe news. We decided to consider newspaper Tweets as negative items for two reasons. Firstly because Spinoza Tweets are about politics and news, thus they deal with topics related to the same domain of Italian daily newspapers. Secondly, because the style of Spinoza Tweets is similar to the style typical of newspapers. Hence Spinoza and newspapers posts have have similar content, similar style, but different intentions.

In order to process the text and build our model we used freely available tools. We used the tokenizer, POS tagger and UKB Word Sense Disambiguation algorithm provided by Freeling (Carreras et al., 2004). We also exploited the Italian WordNet 1.6[2] to get synsets and synonyms, and the sentiment lexicon Sentix[3] (Basile and Nissim, 2013) derived from SentiWordnet (Esuli and Sebastiani, 2006). We used on the CoLFIS Corpus of Written Italian[4] to obtain the usage frequency of a word in written Italian.

4 Method

This section describes two systems: both exploit Decision Trees to classify Tweets as ironic or not. The first system (Section 4.1) is the irony detection approach we propose that relies on lexical and semantic features characterising each word of a Tweet. The second system (Section 4.2) exploits words occurrences (BOW approach) as features useful to train a Decision Tree. The latter system is used as a reference (baseline) to evaluate our irony detection approach.

4.1 Irony Detection Model

Our model for irony detection includes five types of features: Frequency, Synonyms, Ambiguity, Part of Speech, and Sentiments. We included in our model a subset of the features proposed by Barbieri and Saggion (2014a), describing implicit characteristics of each word in a Tweet. We do

[1] The reader can find the list of the Tweet IDs at http://sempub.taln.upf.edu/tw/clicit2014/

[2] http://multiwordnet.fbk.eu/english/home.php

[3] http://www.let.rug.nl/basile/twita/sentix.php

[4] http://linguistica.sns.it/CoLFIS/Home_eng.htm

not consider features such as punctuation, emoticons or number of characters of the Tweet. The proposed features aim to detect two aspects of Tweets that we consider particularly relevant to detect irony: the style used (e.g. register used, frequent or rare words, positive or negative words, etc.) and the unexpectedness in the use of words (Lucariello, 1994) i.e. the presence of "out of context" words (the *gap* feature, see below).

4.1.1 Frequency

We retrieved from the CoLFIS Corpus, the frequency of the word of each Tweet. Thus, we derive three types of Frequency features: *rarest word frequency* (frequency of the most rare word included in the Tweet), *frequency mean* (the arithmetic average of all the frequency of the words in the Tweet) and *frequency gap* (the difference between the two previous features). These features are computed for all the words of each Tweet. We also computed these features by considering only Nouns, Verbs, Adjectives, and Adverbs.

4.1.2 Synonyms

Irony conveys two messages to the audience at the same time, the literal and the intended message (Veale, 2004). We consider the frequencies (in CoLFIS Corpus) of the synonyms of each word in the Tweet, as retrieved from WordNet. Then we compute: the *greatest / lowest number of synonyms* with frequency higher than the one present in the Tweet, the *mean number of synonyms* with frequency greater / lower than the frequency of the related word present in the Tweet. We determine also the greatest / lowest number of synonyms and the mean number of synonyms of the word with frequency greater / lower than the one present in the the Tweet (*gap* feature). We also computed these features separately, considering each of the four POS as before

4.1.3 Ambiguity

Ambiguity plays an important role in irony: a word with more than one meaning can be used to say two (or more) things at the same time. To model the ambiguity of the terms in the Tweets we use the WordNet synsets associated to each word. Our hypothesis is that if a term has many meanings (synsets) it is more likely to be used in an ambiguous way. For each Tweet we calculate the *maximum number of synsets* associated to a single word, the *synset number mean* of all the words, and the *synset gap* that is the difference between the two previous features. We determine the value of these features considering all the words of a Tweet and as well as including only Nouns, Verbs, Adjectives or Adverbs.

4.1.4 Part Of Speech

The features included in the Part Of Speech group are designed to capture the style of the Tweets. The features of this group are eight and each of them counts the number of occurrences of words characterised by a certain POS. The eight POS considered are *Verbs*, *Nouns*, *Adjectives*, *Adverbs*, *Interjections*, *Determiners*, *Pronouns*, and *Adpositions*.

4.1.5 Sentiments

The sentiments of the words in ironic Tweets are important for two reasons: to detect the *sentiment* style (e.g. if ironic Tweets contain mainly positive or negative terms) and to capture unexpectedness created by a negative word in a positive context and viceversa. Relying on Sentix (see Section 3) we compute the *number of positive/negative words*, the *sum of the intensities of the positive/negative words*, the *mean of intensities of positive/negative words*, the *greatest positive/negative score*, the *gap between greatest positive/negative and positive/negative mean*. Then, as before we compute these features for each of the POSs Noun, Verb, Adjetive, and Adverbs.

4.2 Bag Of Word Baseline

Our baseline model is a Decision Tree trained on features represented by the occurrence of the 200 most frequent words in the training set (we calculate the frequent words in each experiment, see Section 5). We only considered words of the message itself, removing expressions such as the name of the newspapers and common patterns like "Continue to read [link]" or "See the Video Gallery on [link]" often present in specific Twitter accounts.

5 Experiments and Results

We obtained from our initial corpus two kinds of datasets: the ironic dataset (that includes all the Tweets from the two Spinoza accounts) and the non-ironic dataset (that is composed by the newspaper Tweets). We choose to classify tweets by a Decision Tree algorithm coupled with the SubsetEvaluation feature selection approach. For our

experiments we used Weka (Witten and Frank, 2005). We train our classifier in a dataset composed of 80% of the Tweets of the ironic dataset and 80% of the Tweets of the non-ironic dataset. The performance of the trained model are tested on a set of Tweets that includes the remaining portions of both ironic and non ironic datasets (20% of each dataset). Examples in the train and test sets are chosen randomly, to avoid correlation of Tweets close in time that are likely to be on the same topic. In addition we run a 10-cross validation using a balanced binary dataset (irony VS one negative topic). We carried out two experiments using the above framework (train/test and 10-cross validation):

1 - We consider as positive examples the ironic Tweets from Spinoza, and as negative examples each Tweet of the seven newspapers (this experiment is performed seven times, as we compare irony with each newspaper).

2 - We consider as positive example the ironic Tweets from Spinoza as before, while the negative dataset includes Tweets from all the seven newspaper (each newspaper contributes with a number of Tweets equal to 455).

We run the two experiments using both our feature set for irony detection (Section 4.1) and the BOW baseline features (Section 4.2). The results are reported in Table 1, organised in Precision, Recall and F-Measure.

6 Discussion

Our system always outperforms the BOW baseline. In Experiment 1 (irony versus each newspaper) our model outperforms the BOW approach by at least 4 points (F1). In Experiment 2 (irony versus a composition of all the newspapers) the results of BOW are still worse, six points less, and not due by chance (according to the McNemar's statistical test, 0.01 significance level). Moreover, in Experiment 2 the BOW baseline obtains its worst result, suggesting that this approach models the style of a specific newspaper rather than the ironic Tweets. On the other hand our system seems to better adapt to this situation indicating that it is less influenced by the non-ironic examples (a good characteristic as in a realistic case the non-ironic examples are unknown and of any type). The best features (information gain 0.20/0.15) are number of verbs and synset related features (Ambiguity, Section 4.1.3).

		Test Set			10-Folds		
	Data	**P**	**R**	**F**	**P**	**R**	**F**
Bag Of Words	Corr	.74	.68	.71	.72	.69	.70
	Gazz	.67	.70	.69	.71	.70	.70
	Mess	.71	.66	.68	.71	.67	.69
	Repu	.72	.68	.70	.70	.67	.69
	Rest	.77	.70	.73	.76	.72	.74
	Sol24	.71	.71	.71	.70	.70	.70
	Stam	.73	.66	.64	.70	.64	.66
	MIX	.69	.62	.65	.70	.61	.65
Our Model	Corr	.77	.76	.76	.78	.73	.75
	Gazz	.77	.76	.76	.75	.75	.75
	Mess	.73	.72	.72	.71	.70	.70
	Repu	.80	.75	.77	.73	.73	.73
	Rest	.87	.77	.82	.80	.78	.79
	Sol24	.76	.79	.78	.74	.72	.73
	Stam	.74	.75	.75	.74	.73	.72
	MIX	**.75**	**.76**	**.76**	.72	.70	.71

Table 1: Precision, Recall and F-Measure of each run of Experiment 1 and Experiment 2 ("MIX")

7 Conclusion and Future Work

In this study we evaluate a novel system to detect irony in Italian, focusing on Tweets. We tackle this problem as binary classification, where the ironic examples are posts of the Twitter account Spinoza and the non-ironic examples are Tweets from seven popular Italian newspapers. We evaluated the effectiveness of Decision Trees with different feature sets to carry out this classification task. Our system only focuses on characteristics on lexical and semantic information that characterises each word, rather than the words themselves as features. The performance of the system is good if compared to our baseline (BOW) considering only word occurrences as features, since we obtain an F1 improvement of 0.11. This result shows the suitability of our approach to detect ironic Italian Tweets. However, there is space to enrich and tune the model as this is only a first approach. It is possible to both improve the model with new features (for example related to punctuation or language models) and evaluate the system on new and extended corpora of Italian Tweets as they become available. Another issue we faced is the lack of accurate evaluations of features performance considering distinct classifiers / algorithms for irony detection.

References

Francesco Barbieri and Horacio Saggion. 2014a. 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, Gothenburg, Sweden, April. Association for Computational Linguistics.

Francesco Barbieri and Horacio Saggion. 2014b. Modelling Irony in Twitter, Features Analysis and Evaluation. In *Language Resources and Evaluation conference, LREC*, Reykjavik, Iceland, May.

Valerio Basile and Malvina Nissim. 2013. Sentiment analysis on italian tweets. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 100–107.

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

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

Xavier Carreras, Isaac Chao, Lluis Padró, and Muntsa Padró. 2004. Freeling: An open-source suite of language analyzers. In *LREC*.

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.

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. Association for Computational Linguistics.

Larissa A de Freitas, Aline A Vanin, Denise N Hogetop, Marco N Bochernitsan, and Renata Vieira. 2014. Pathways for irony detection in tweets. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pages 628–633. ACM.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of Language Resources and Evaluation Conference*, volume 6, pages 417–422.

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, Istanbul, Turkey*, pages 1–7.

Rachel Giora. 1995. On irony and negation. *Discourse processes*, 19(2):239–264.

Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying Sarcasm in Twitter: A Closer Look. In *ACL (Short Papers)*, pages 581–586. Citeseer.

H Paul Grice. 1975. Logic and conversation. *1975*, pages 41–58.

Christine Liebrecht, Florian Kunneman, and Antal van den Bosch. 2013. The perfect solution for detecting sarcasm in tweets# not. *WASSA 2013*, page 29.

Joan Lucariello. 1994. Situational irony: A concept of events gone awry. *Journal of Experimental Psychology: General*, 123(2):129.

Quintilien and Harold Edgeworth Butler. 1953. *The Institutio Oratoria of Quintilian. With an English Translation by HE Butler*. W. Heinemann.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in Twitter. *Language Resources and Evaluation*, pages 1–30.

Akira Utsumi. 2000. Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony. *Journal of Pragmatics*, 32(12):1777–1806.

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

Tony Veale and Yanfen Hao. 2010b. Detecting Ironic Intent in Creative Comparisons. In *ECAI*, volume 215, pages 765–770.

Tony Veale. 2004. The challenge of creative information retrieval. In *Computational Linguistics and Intelligent Text Processing*, pages 457–467. Springer.

Deirdre Wilson and Dan Sperber. 2002. Relevance theory. *Handbook of pragmatics*.

Ian H Witten and Eibe Frank. 2005. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Ugan Yasavur, Jorge Travieso, Christine Lisetti, and Naphtali Rishe. 2014. Sentiment analysis using dependency trees and named-entities. In *The Twenty-Seventh International Flairs Conference*.