Automatic Detection of Irony and Humour in Twitter

Francesco Barbieri
Pompeu Fabra University
Barcelona, Spain
francesco.barbieri@upf.edu

Horacio Saggion
Pompeu Fabra University
Barcelona, Spain
horacio.saggion@upf.edu

Abstract

Irony and humour are just two of many forms of figurative language. Approaches to identify in vast volumes of data such as the internet humorous or ironic statements is important not only from a theoretical view point but also for their potential applicability in social networks or human-computer interactive systems. In this study we investigate the automatic detection of irony and humour in social networks such as Twitter casting it as a classification problem. We propose a rich set of features for text interpretation and representation to train classification procedures. In cross-domain classification experiments our model achieves and improves state-of-the-art performance.

Introduction

Irony and humour are just two examples of figurative language (Reyes, Rosso, and Veale 2013). Approaches to identify in vast volumes of data such as the internet humorous or ironic statements are important not only from a theoretical view point but also for their potential applicability in social network analysis and human-computer interactive systems. Systems able to select humorous/ironic statements on a given topic to present to a user are important in human-machine communication. It is also important for a system being able to recognise when users are being ironic/humorous to appropriate deal with their requests. Irony has also relevance in the field of sentiment analysis and opinion mining (Pang and Lee 2008) since it can be used to express a negative statement in an apparently positive way. However, irony detection appears as a difficult problem since ironic statements are used to express the contrary of what is being said (Quintilien and Butler 1953), therefore being a tough nut to crack by current systems. Reyes et al. (2013) approach the problem as one of classification training machine learning algorithms to sepatate ironic from non-ironic statements. Humour has been studied for a number of years in computational linguistics in terms of both humour generation (Stock and Strapparava 2006; Ritchie and Masthoff 2011) and interpretation (Mihalcea and Pulman 2007; Taylor and Mazlack 2005). In particular it has also been approached as classification by Mihalcea and Strapparava (2005) creating a specially designed corpus of one-liners (i.e., one sentence jokes) as the positive class and headlines and other short statements as a negative class.

Following these lines of research, we first try to detect these topics separately; then, since they are both figurative language, and they may have some correlation, we also try to detect them at the same time (we use the union of them as positive example). This last experiment is interesting as it will give us hints for figurative language detection, hence it will help us exploring new aspects of creativity in language (Veale and Hao 2010b). This experiment can be seen as a small step toward the design of a machine capable to evaluate creativity, and with further work also capable to generate creative utterances.

Our dataset is composed of text retrieved from the microblogging service Twitter[1].

For the experiments to be presented in this paper we use a dataset created for the study of irony detection which allows us to compare our findings with recent state-of-the-art approaches (Reyes, Rosso, and Veale 2013). The dataset also contains humorous tweets therefore being appropriate for our purpose.

The contributions of this paper are as follows:

- the evaluation of our irony detection model (Barbieri and Saggion 2014) to humour classification;
- a comparison of our model with the state-of-the-art; and
- a novel set of experiments to demonstrate cross-domain adaptation.

The paper will show that our model achieves and improve state-of-the-art performance, and that it can be applied to different domains.

Related Work

Verbal irony has been defined in several ways over the years but there is no consensual agreement on its definition. The standard definition is considered "saying the opposite of what you mean" (Quintilien and Butler 1953) where the opposition of literal and intended meanings is very clear. Grice (1975) believes that irony is a rhetorical figure that violates the maxim of quality: "Do not say what you believe to be false". Irony is also defined (Giora 1995) as any form of negation with no negation markers (as most

[1] https://twitter.com/

of the ironic utterances are affirmative, and ironic speakers use indirect negation). Wilson and Sperber (2002) defined it as echoic utterance that shows a negative aspect of someone's else opinion. Finally irony has been defined as form of pretence by Utsumi (2000) and Veale and Hao (2010b). Veale states that "ironic speakers usually craft their utterances in spite of what has just happened, not because of it. The pretence alludes to, or echoes, an expectation that has been violated". Past computational approaches to irony detection are scarce. Carvalho et. al (2009) created an automatic system for detecting irony relying on emoticons and special punctuation. They focused on detection of ironic style in newspaper articles. Veale and Hao (2010a) proposed an algorithm for separating ironic from non-ironic similes, detecting common terms used in this ironic comparison. Reyes et. al (2013) have recently proposed a model to detect irony in Twitter, which is based on four groups of features: signatures, unexpectedness, style, and emotional scenarios. Their classification results support the idea that textual features can capture patterns used by people to convey irony. Among the proposed features, *skip-grams* (part of the style group) which captures word sequences that contain (or skip over) arbitrary gaps, seems to be the best one. Computational approaches to humour generation include among others the JAPE system (Ritchie 2003) and the STANDUP riddle generator program (Ritchie and Masthoff 2011) which are largely based on the use of a dictionary for humorous effect. It has been argued that humorous discourse depend on the fact that they can have multiple interpretations, that is they are ambiguous. These characteristics are explored in approaches to humour detection. Mihalcea and Strappavara (2005) study classification of a restricted type of humorous discourse: *one-liners*, which have the purpose of producing humorous effect in very few words. They created a dataset semi-automatically by retrieving itemized sentences from web sites whose URLs contain words such as "oneliner", "humour", "joke", etc. Non-humorous data was created using Reuters titles, Proverbs, and sentences extracted from the British National Corpus. They use two types of models to separate humorous from non-humorous texts. On the one hand a specially designed set of features is created to model *Alliteration*, *Antonymy*, and *Slang* of a sexual oriented nature. On the other hand they tried a word-based text classification algorithm. Non surprisingly the word-based classifier is much more effective than the specially designed features. In (Mihalcea and Pulman 2007) additional features to model violated expectations, human oriented activities, and polarity are introduced. Veale (2013) also created a dataset of humorous similes by querying the web with specific similes patterns.

Data and Text Processing

The dataset used for the experiments reported in this paper has been prepared by Reyes et al. (2013). It is a corpus of 40.000 tweets equally divided into four different topics: *Irony*, *Education*, *Humour*, and *Politics*. The tweets were automatically selected by looking at Twitter hashtags (#irony, #education, #humour, and #politics) added by users in order to link their contribution to a particular subject and community. The hashtags are removed from the tweets for the experiments. According to Reyes et. al (2013), these hashtags were selected for three main reasons: (i) to avoid manual selection of tweets, (ii) to allow irony analysis beyond literary uses, and because (iii) irony hashtag may reflect a tacit belief about what constitutes irony and humour.

Another corpora is employed in our approach to measure the frequency of word usage. We adopted the Second Release of the American National Corpus Frequency Data[2] (Ide and Suderman 2004), which provides the number of occurrences of a word in the written and spoken ANC. From now on, we will mean with "frequency of a term" the absolute frequency the term has in the ANC.

In order to process the tweets we used the Gate plugin *Twitie* (Bontcheva et al. 2013), an open-source information extraction pipeline for Microblog Text. We used it as tokeniser and part-of-speech tagger. We also adopted Rita WordNet API (Howe 2009) and Java API for WordNet Searching (Spell 2009) to perform operations on WordNet synsets (Miller 1995).

Methodology

We approach the detection of irony and humour as a classification problem applying supervised machine learning methods to the Twitter corpus previously introduced. When choosing the classifiers we had avoided those requiring features to be independent (e.g. Naive Bayes) as some of our features are not. Since we approach the problem as a binary decision (deciding if a tweet is ironic or not) we picked two tree-based classifiers: Random Forest and Decision tree (the latter allows us to compare our findings directly to Reyes et. al (2013)). We use the implementations available in the Weka toolkit (Witten and Frank 2005).

To represent each tweet we use seven groups of features. Some of them are designed to detect imbalance and unexpectedness, others to detect common patterns in the structure of the tweets (like type of punctuation, length, emoticons). Below is an overview of the group of features in our model:

- Frequency *(gap between rare and common words)*
- Written-Spoken *(written-spoken style uses)*
- Intensity *(intensity of adverbs and adjectives)*
- Structure *(length, punctuation, emoticons, links)*
- Sentiments *(gap between positive and negative terms)*
- Synonyms *(common vs. rare synonyms use)*
- Ambiguity *(measure of possible ambiguities)*

In the following sections we describe the theoretical motivations behind the features and how them have been implemented.

[2]The American National Corpus (http://www.anc.org/) is, as we read in the web site, a massive electronic collection of American English words (15 million)

Frequency

Unexpectedness and Incongruity can be a signals of irony and humour (Lucariello 2007; Venour 2013). In order to study these aspects we explore the frequency imbalance between words, i.e. register inconsistencies between terms of the same tweet. The intuition is that the use of many words commonly used in English (i.e. high frequency in ANC) and only a few terms rarely used in English (i.e. low frequency in ANC) in the same sentence creates imbalance that may cause unexpectedness, since within a single tweet only one kind of register is expected. We are able to explore this aspect using the ANC Frequency Data corpus.

Three features belong to this group: **frequency mean**, **rarest word**, **frequency gap**. The first one is the arithmetic average of all the frequencies of the words in a tweet, and it is used to detect the *frequency style* of a tweet. The second one, **rarest word**, is the frequency value of the rarest word, designed to capture the word that may create imbalance.

Written-Spoken

Twitter is composed of written text, but an informal spoken English style is often used. We designed this set of features to explore unexpectedness and incongruity created by using spoken style words in a mainly written style tweet or vice versa (formal words usually adopted in written text employed in a spoken style context). We can analyse this aspect with ANC written and spoken, as we can see using this corpora whether a word is more often used in written or spoken English. There are three features in this group: **written mean**, **spoken mean**, **written spoken gap**. The first and second ones are the means of the frequency values, respectively, in written and spoken ANC corpora of all the words in the tweet. The third one, **written spoken gap**, is the absolute value of the difference between the first two, designed to see if ironic writers use both styles (creating imbalance) or only one of them. A low difference between written and spoken styles means that both styles are used.

Structure

With this group of features we want to study the structure of the tweet: if it is long or short (length), if it contains long or short words (mean of word length), and also what kind of punctuation is used (exclamation marks, emoticons, etc.). This is a powerful feature, as ironic and humorous tweets in our corpora present specific structures: for example ironic tweets are longer (mean length of an ironic tweet is 94.7 characters against 82.0467, 86.5776, 86.5307 of the other topics), and humorous tweets use more emoticons than the other domains (mean number of emoticons in a humorous tweet is 0.012 and in the other corpora is only 0.003, 0.001, 0.002). The Structure group includes several features that we describe below.

The **length** feature consists of the number of characters that compose the tweet, **n. words** is the number of words, and **words length mean** is the mean of the words length. Moreover, we use the number of verbs, nouns, adjectives and adverbs as features, naming them **n. verbs**, **n. nouns**, **n. adjectives** and **n. adverbs**. With these last four features we also computed the ratio of each part of speech to the number of words in the tweet; we called them **verb ratio**, **noun ratio**, **adjective ratio**, and **adverb ratio**. All these features have the purpose of capturing the style of the writer.

Inspired by Davidov et al. (2010) and Carvalho (2009) we designed features related to punctuation. These features are: number of **commas**, **full stops**, **ellipsis**, **exclamation** and **quotation** marks that a tweet contain.

We also added the feature **laughs** which is the number of *hahah*, *lol*, *rofl*, and *lmao*.

Additionally, there are the *emoticon* feature, that is the number of *:)*, *:D*, of *:(*, and *;)* in a tweet. This feature works well in the Humour corpus as it contains four times more emoticons than the other corpora. The ironic corpus is the one with the least emoticons (there are only 360 emoticons in the Irony corpus, while in Humour, Education, and Politics tweets they are 2065, 492, 397 respectively). In the light of these statistics we can argue that ironic authors avoid emoticons and leave words to be the central thing: the audience has to understand the irony without explicit signs, like emoticons. Humour seems, on the other hand, more explicit.

Finally we added a simple but powerful feature, *web-links*. It simply say if a tweet include or not an internet link. This feature result good for Humour and excellent for Irony, where internet links are not used frequently.

Intensity

We also study the intensity of adjectives and adverbs. We adopted the intensity scores of Potts (2011) who uses naturally occurring metadata (star ratings on service and product reviews) to construct adjectives and adverbs scales. An example of adjective scale (and relative scores in brackets) could be the following: horrible (-1.9) $\rightarrow$ bad (-1.1) $\rightarrow$ good (0.2) $\rightarrow$ nice (0.3) $\rightarrow$ great (0.8).

With these scores we evaluate four features for adjective intensity and four for adverb intensity (implemented in the same way): **adj (adv) tot**, **adj (adv) mean**, **adj (adv) max**, and **adj (adv) gap**. The sum of the AdjScale scores of all the adjectives in the tweet is called **adj tot**. **adj mean** is **adj tot** divided by the number of adjectives in the tweet. The maximum AdjScale score within a single tweet is **adj max**. Finally, **adj gap** is the difference between **adj max** and **adj mean**, designed to see "how much" the most intense adjective is out of context.

Synonyms

As previously said, irony convey two messages to the audience at the same time (Veale 2004). It follows that the choice of a term (rather than one of its synonyms) is very important in order to send the second, not obvious, message. The choice of the synonym is an important feature for humour as well, and it seems that authors of humours tweets prefer using common terms.

For each word of a tweet we get its synonyms with WordNet (Miller 1995), then we calculate their ANC frequencies and sort them into a decreasing ranked list (the actual word is part of this ranking as well). We use these rankings to define the four features which belong to this group. The first

one is **syno lower** which is the number of synonyms of the word w_i with frequency lower than the frequency of w_i. It is defined as in Equation 1:

$$sl_{w_i} = |syn_{i,k} \ : \ f(syn_{i,k}) < f(w_i)| \quad (1)$$

where $syn_{i,k}$ is the synonym of w_i with rank k, and $f(x)$ the ANC frequency of x. Then we also defined **syno lower mean** as mean of sl_{w_i} (i.e. the arithmetic average of sl_{w_i} over all the words of a tweet).

We also designed two more features: **syno lower gap** and **syno greater gap**, but to define them we need two more parameters. The first one is *word lowest syno* that is the maximum sl_{w_i} in a tweet. It is formally defined as:

$$wls_t = \max_{w_i}\{|syn_{i,k} \ : \ f(syn_{i,k}) < f(w_i)|\} \quad (2)$$

The second one is *word greatest syno* defined as:

$$wgs_t = \max_{w_i}\{|syn_{i,k} \ : \ f(syn_{i,k}) > f(w_i)|\} \quad (3)$$

We are now able to describe **syno lower gap** which detects the imbalance that creates a common synonym in a context of rare synonyms. It is the difference between *word lowest syno* and **syno lower mean**. Finally, we detect the gap of very rare synonyms in a context of common ones with **syno greater gap**. It is the difference between *word greatest syno* and *syno greater mean*, where *syno greater mean* is the following:

$$sgm_t = \frac{|syn_{i,k} \ : \ f(syn_{i,k}) > f(w_i)|}{n.\ words\ of\ t} \quad (4)$$

Ambiguity

Another interesting aspect of irony and humour is ambiguity. We noticed that ironic tweets includes the greatest arithmetic average of the number of WordNet synsets, and humour the least; this indicates that ironic tweets presents words with more meanings, an humorous tweets words with less meaning. In the case of irony, our assumption is that if a word has many meanings the possibility of "saying something else" with this word is higher than in a term that has only a few meanings, then higher possibility of sending more then one message (literal and intended) at the same time.

There are three features that aim to capture these aspects: **synset mean**, **max synset**, and **synset gap**. The first one is the mean of the number of synsets of each word of the tweet, to see if words with many meanings are often used in the tweet. The second one is the greatest number of synsets that a single word has; we consider this word the one with the highest possibility of being used ironically (as multiple meanings are available to say different things). In addition, we calculate **synset gap** as the difference between the number of synsets of this word (**max synset**) and the average number of synsets (**synset mean**), assuming that if this gap is high the author may have used that inconsistent word intentionally.

Sentiments

We analyse also the sentiments of irony and humour by using the SentiWordNet sentiment lexicon (Esuli and Sebastiani 2006) that assigns to each synset of WordNet sentiment scores of positivity and negativity.

There are six features in the Sentiments group. The first one is named **positive sum** and it is the sum of all the positive scores in a tweet, the second one is **negative sum**, defined as sum of all the negative scores. The arithmetic average of the previous ones is another feature, named **positive negative mean**, designed to reveal the sentiment that better describe the whole tweet. Moreover, there is **positive-negative gap** that is the difference between the first two features, as we wanted also to detect the positive/negative imbalance within the same tweet.

The imbalance may be created using only one single very positive (or negative) word in the tweet, and the previous features will not be able to detect it, thus we needed to add two more. For this purpose the model includes **positive single gap** defined as the difference between most positive word and the mean of all the sentiment scores of all the words of the tweet and **negative single gap** defined in the same way, but with the most negative one.

Experiments and Results

The experiments described in this section aim at verifying: (i) the discriminative power of our model, (i) the portability of the model across domains, and (iii) its state-of-the-art status. In order to carry out experimentation and to be able to compare our approach to that of (Reyes, Rosso, and Veale 2013) we use several datasets derived from the corpus used in the paper.

Irony Detection

Our first experiment addresses the problem of irony detection comparing the performance of our model with that of Reyes et al. (Reyes, Rosso, and Veale 2013). In order to replicate their experimental setting, three balanced datasets were created from the corpus: (i) *Irony vs Humour*, (ii) *Irony vs Education*, and (iii) *Irony vs Politics*. Each dataset is composed of 10,000 examples of irony and 10,000 examples of a different topic. A 10-fold cross-validation experiment was run in each dataset and precision, recall, and f-measure computed. The results of the experiments are presented in Table 1.

Cross-domain Irony and Humour Detection

Our second experiment addresses cross-domain adaptation, which has not been addressed in previous work. We designed three balanced *training* sets composed of 7500 positive tweets (irony or humour) and 7500 of each negative topic that remain available (Education/Humour/Politics when the positive is Irony and Education/Irony/Politics when the positive is Humour) and three balanced *test* sets composed of 2500 positive and 2500 of each negative topic (Education/Humour/Politics when the positive is Irony and Education/Irony/Politics when the positive is Humour). We carried out all the Train/Test possible combinations to verify

	Education			Humour			Politics		
Model	P	R	F1	P	R	F1	P	R	F1
Reyes et. al	.76	.66	.70	.78	.74	.76	.75	.71	.73
Our model	.87	.87	**.87**	.88	.88	**.88**	.87	.87	**.87**

Table 1: Precision, Recall, and F-Measure over the three corpora Education, Humour, and Politics. Both our and Reyes et al. results are shown; the classifier used is Decision Tree for both models.

	Training Set								
	Education			**Humour**			**Politics**		
Test set	P	R	F1	P	R	F1	P	R	F1
Education	.87/.89	.87/.89	.87/.89	.86/.86	.86/.85	.86/.85	.86/.87	.86/.87	.86/.87
Humour	.78/.79	.77/.74	.77/.74	.88/.89	.88/.89	.88/.89	.78/.79	.77/.74	.76/.74
Politics	.82/.83	.82/.83	.82/.82	.83/.83	.82/.82	.82/.82	.88/.89	.88/.89	.88/.89

Table 2: Results of Experiment 2 when positive topic is Irony and negative topics are Education, Humour and Politics. The table includes Precision, Recall and F-Measure for each Training/Testing topic combination written in the form "Decision Tree / Random Forest" as we used these two algorithms as classifiers.

how the model works when the domain is changed (one such instance is to train in the Irony/Politics dataset and evaluate it in the Irony/Education dataset). The results of the experiments are presented in Tables 2 and 3.

Figurative Language Filtering

Our third experiment consists on treating irony and humour as a single class representing figurative language; here we want to verify whether our model can separate "figurative" from "non-figurative" language. We designed one balanced Training set composed of 15000 positive tweets (7500 of Irony and 7500 of Humour) and 15000 negative examples (7500 of Education and 7500 of Politics). Then a balanced Test set composed of 5000 positive tweets (2500 of Irony and 2500 of Humour) and 5000 negative examples (2500 of Education and 2500 of Politics). Table 4 presents results of this experiment comparing two classification algorithms: Decision Tree and Random Forest.

Feature Analysis

Finally and in order to have a clear understanding about the contribution of each features of our model, we also studied the behaviour of information gain in each dataset. We compute information gain experiments over the three Training sets of our "cross-domain" experiments. Information gain results are directly correlated to the classification results as we are using tree based classifiers and features with high information gain will be at the top of the tree i.e. important discriminators. Figure 1 shows the information gain when the positive topic is Irony, Figure 2 when the positive topic is Humour. In Table 5 (a) and (b) are shown the Pearson Correlation between information gain of each feature over different topics when training Irony and Humour. The correlation has been calculated to determine whether the system uses similar features for different negative topics (if the correlation is low we are likely to have cross-domain problems). The correlation can tell us how well correlated two topics are.

Discussion

Looking at the figures obtained in our irony detection experiments, it appears that our model is more balanced in terms of precision and recall and that our overall f-measure improves over previous work having the additional advantage of the features being easy to compute.

Now turning to the cross-domain experiments we observe that our model performs reasonably well across-domains. That is to say except when we try to identify humorous tweets having trained with irony. This is in fact an interesting result which may indicate that not all features of our model are appropriate for humorous discourse, requiring the design of additional features for this type of figurative language.

With respect to the figurative language filtering experiments, results seem promising. Our experiments can not be compared with previous approaches directly because of differences in datasets but we point out that in humour classification (Mihalcea and Strapparava 2005) using specially designed "humour" characteristics accuracy results are around 76%.

Finally, our feature analysis experiments (Figures 1 and 2), we observe that features for structure, frequency, and synonymy are discriminators of irony. Although there is great variability across domains which is also shown in the correlation Table 5. Where humour is concerned, we see that features of structure, synonymy, frequency and intensity also are good discriminators again with great variability across domains. Features belonging to ambiguity and sentiment have little discriminative power. Regarding figurative versus not figurative experiment the best features are **syno lower**, **rarest val**, **word length** and **adj/adv max**. In comparison to education and politics, humour and irony include longer (**word length**) and more common words (**syno lower**, **rarest val**). Moreover, intensity of adjectives and adverbs (**adj/adv max**) is important characteristic as humour and irony include more intense terms.

	Training Set								
	Education			**Irony**			**Politics**		
Test set	P	R	F1	P	R	F1	P	R	F1
Education	.78/.81	.78/.81	.78/.81	.55/.57	.53/.53	.46/.43	.72/.77	.71/.75	.71/.75
Irony	.72/.64	.71/.61	.71/.58	.88/.89	.88/.88	.88/.88	.60/.67	.69/.63	.69/.61
Politics	.73/.77	.73/.76	.73/.76	.60/.61	.56/.55	.51/.48	.80/.84	.80/.84	.80/.84

Table 3: Results of Experiment 2 when positive topic is Humour and negative topics are Education, Irony and Politics. The table includes Precision, Recall and F-Measure for each Training/Testing topic combination written in the form "Decision Tree / Random Forest" as we used these two algorithms for the classifications.

P	R	F1
.80/.83	.80/.83	.80/.83

Table 4: Figurative language filtering results. Precision, Recall, and F-measure numbers correspond to two algorithms: Decision Tree/Random Forest.

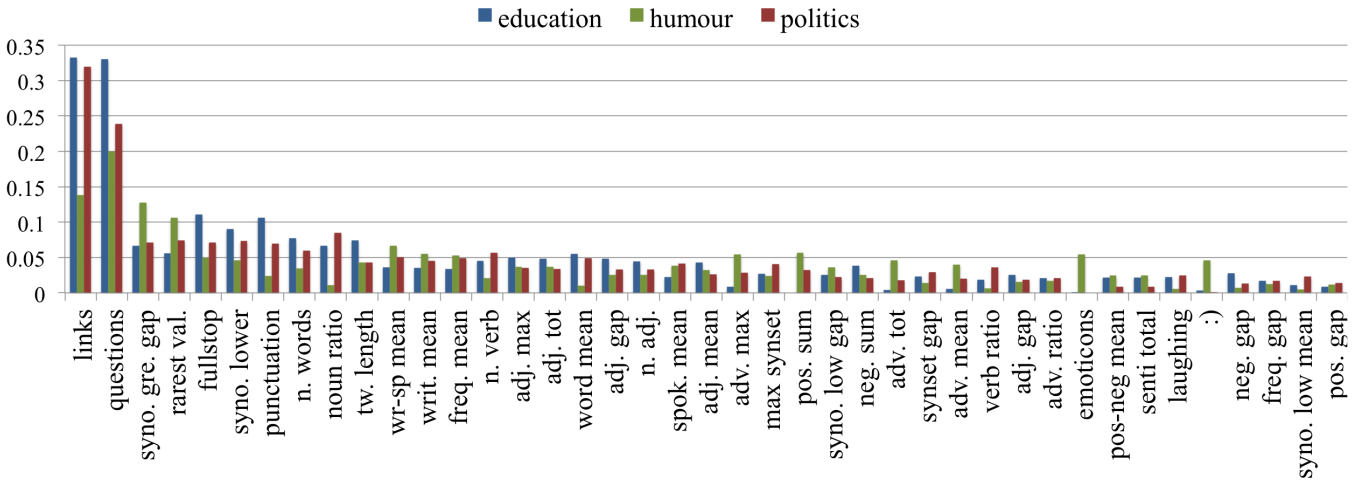

Figure 1: Information gain of each feature of the model. Irony corpus is compared to Education, Humour, and Politics corpora. High values of information gain help to better discriminate ironic from non-ironic tweets.

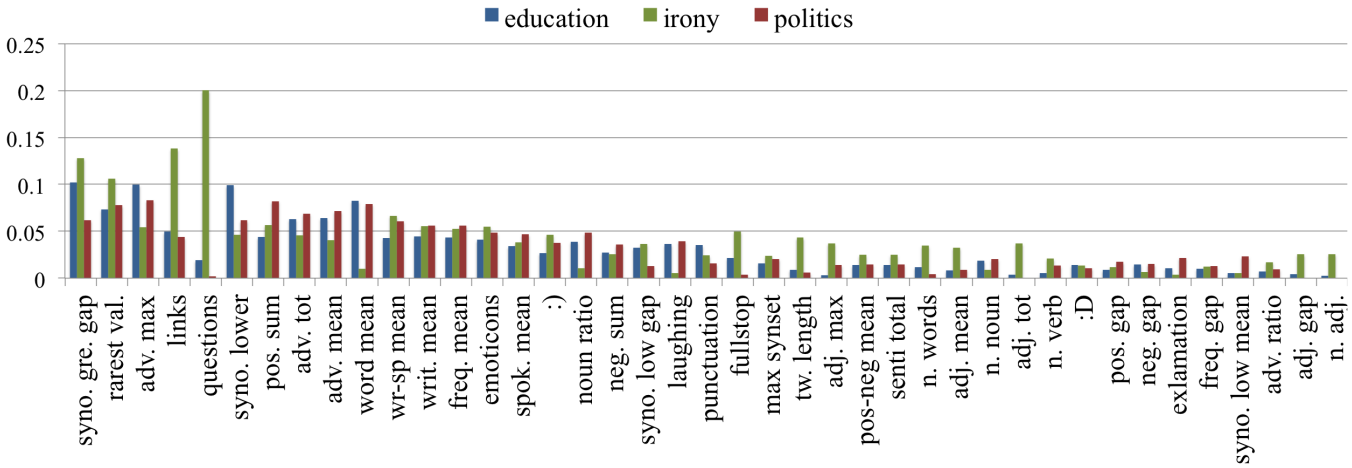

Figure 2: Information gain of each feature of the model. Humour corpus is compared to Education, Irony, and Politics corpora. High values of information gain help to better discriminate humorous from non-humorous tweets.

(a)	Education	Humour	Politics
Education	1	0.76	0.96
Humour	-	1	0.76
Politics	-	-	1

(b)	Education	Irony	Politics
Education	1	0.48	0.89
Irony	-	1	0.36
Politics	-	-	1

Table 5: Pearson Correlation between information gain of each feature over different topics when training on Irony (a) or Humour (b)

Conclusion and Future Work

In this article we have proposed a novel linguistically motivated set of features to detect irony and humour in the social network Twitter. The features take into account frequency, written/spoken differences, sentiments, ambiguity, intensity, synonymy and structure. We have designed many of them to be able to model "unexpectedness" and "incongruity", a key characteristic of both genres.

We have performed controlled experiments with an available corpus used in previous work which allow us to carried out experimentation in different scenarios. First, we carried out experiments to verify the performance of our set of features compared with previous work obtaining promising results. Second, we have carried out cross-domain experiments to show that the model can be used across domains. This experiment also shows that additional features are needed because irony and humour have their own particular characteristics. Third, we have performed an experiment to try to classify figurative language obtaining initial reasonable results. There is however much space for improvements. The ambiguity aspect is still weak in this research, and it needs to be improved. Also experiments adopting different topics may be useful in order to explore the system behaviour in a more realistic situation. We plan to model additional features to better distinguish between the two forms of figurative language.

Acknowledgments

We are grateful to three anonymous reviewers for their comments and suggestions that help improve our paper. The research described in this paper is partially funded by fellowship RYC-2009-04291 from Programa Ramón y Cajal 2009 and project number TIN2012-38584-C06-03 (SKATER-UPF-TALN) from Ministerio de Economía y Competitividad, Secretaría de Estado de Investigación, Desarrollo e Innovación, Spain. We also acknowledge partial support from the EU project Dr. Inventor (FP7-ICT-2013.8.1 project number 611383).

References

Barbieri, F., and Saggion, H. 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*, 56–64. Gothenburg, Sweden: Association for Computational Linguistics.

Bontcheva, K.; Derczynski, L.; Funk, A.; Greenwood, M. A.; Maynard, D.; and Aswani, N. 2013. Twitie: An open-source information extraction pipeline for microblog text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*. Association for Computational Linguistics.

Carvalho, P.; Sarmento, L.; Silva, M. J.; and de Oliveira, E. 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*, 53–56. ACM.

Davidov, D.; Tsur, O.; and Rappoport, A. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, 107–116. Association for Computational Linguistics.

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

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

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

Howe, D. C. 2009. Rita wordnet. java based api to access wordnet.

Ide, N., and Suderman, K. 2004. The American National Corpus First Release. In *Proceedings of the Language Resources and Evaluation Conference*.

Lucariello, J. 2007. Situational irony: A concept of events gone away. *Irony in language and thought* 467–498.

Mihalcea, R., and Pulman, S. G. 2007. Characterizing humour: An exploration of features in humorous texts. In *CICLing*, 337–347.

Mihalcea, R., and Strapparava, C. 2005. Making computers laugh: Investigations in automatic humor recognition. In *HLT/EMNLP*.

Miller, G. A. 1995. Wordnet: a lexical database for english. *Communications of the ACM* 38(11):39–41.

Pang, B., and Lee, L. 2008. Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.* 2(1-2):1–135.

Potts, C. 2011. Developing adjective scales from user-supplied textual metadata. *NSF Workshop on Restructuring Adjectives in WordNet. Arlington,VA*.

Quintilien, and Butler, H. E. 1953. *The Institutio Oratoria of Quintilian. With an English Translation by HE Butler*. W. Heinemann.

Reyes, A.; Rosso, P.; and Veale, T. 2013. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation* 1–30.

Ritchie, G., and Masthoff, J. 2011. The STANDUP 2 interactive riddle builder. In Ventura, D.; Gervás, P.; Harrell, D. F.; Maher, M. L.; Pease, A.; and Wiggins, G., eds., *Proceedings of the Second International Conference on Computational Creativity*, 159.

Ritchie, G. 2003. The jape riddle generator: technical specification. Technical report, University of Edingurgh.

Spell, B. 2009. Java api for wordnet searching (jaws).

Stock, O., and Strapparava, C. 2006. Laughing with hahacronym, a computational humor system. In *AAAI*, 1675–1678.

Taylor, J., and Mazlack, L. 2005. Toward computational recognition of humorous intent. In *Cognitive Science Conference*.

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

Veale, T., and Hao, Y. 2010a. Detecting ironic intent in creative comparisons. In *ECAI*, volume 215, 765–770.

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

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

Veale, T. 2013. Humorous similes. *Humor* 26(1):3–22.

Venour, C. 2013. *A computational model of lexical incongruity in humorous text*. Ph.D. Dissertation, University of Aberdeen.

Wilson, D., and Sperber, D. 2002. Relevance theory. *Handbook of pragmatics*.

Witten, I. H., and Frank, E. 2005. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.