

Character and Word Baselines for Irony Detection in Spanish Short Texts*

Sistemas de detección de ironía basados en palabras y caracteres para textos cortos en español

Gabriela Jasso, Ivan Meza

Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas
Universidad Nacional Autónoma de México
Ciudad Universitaria
{gabriela,ivanvladimir}@turing.iimas.unam.mx

Resumen: La ironía verbal es un fenómeno lingüístico en donde el significado expresado es el opuesto al significado literal del mensaje. Es un reto para el Procesamiento de Lenguaje Natural ya que se debe enseñar a un sistema una forma de reconocer y procesar el cambio de polaridad de lo expresado. Aún cuando han habido esfuerzos recientes en la identificación de ironía y sarcasmo, ninguno de estos aborda el problema en español. En este trabajo nos enfocamos en establecer un sistema base de clasificación usando características simples al nivel de palabras y caracteres para entradas en español de la red social *Twitter*. Presentamos sistemas basados en *máquinas de soporte vectorial* y *selvas aleatorias usando n-gramas*, así como un enfoque distribucional (i.e., *word2vec*).

Palabras clave: Detección de ironía, ironía verbal, textos cortos, word2vec

Abstract: Verbal irony is the linguistic phenomenon in which the expressed meaning is the opposite of the literal meaning. Irony is a challenging task for Natural Language Processing, since one must teach a system to identify and process the polarity of the expression. Although there have been recent efforts in irony and sarcasm identification, none of them tackle the issue in Spanish. In this work we focus on producing classification baseline systems using straight-forward word and character features for Spanish posts from the social network *Twitter*. We present a set of *n*-gram baselines using *support vector machines* and *random forests* classifiers, as well as for a distribucional approach (i.e., *word2vec*).

Keywords: Irony detection, verbal irony, short text, word2vec

1 Introduction

Irony is a non-literal phenomenon that has been widely studied. Haverkate proposes three types of irony: dramatic, situational and verbal (Haverkate, 1990). Dramatic and situational irony describe contradictory events and their relation in a discourse while verbal irony concentrates only on the discourse. The detection of verbal irony expressions is of particular interest to Natural Language Processing because they tend to express the opposite to their literal meaning. The correct detection of such expressions is challenging since there is no evident marker

that an expression is “ironic” and most of the time they use the same superficial forms than those of a non-ironic expression. However, its detection has an impact on tasks which highly depend on the polarity of the meaning such as sentiment analysis. In this work we focus on the detection of verbal irony.

Another difficulty on capturing irony in a computational system is its association with sarcasm and satire, other non-literal phenomenon (Reyes, Rosso, and Buscaldi, 2012). In this work we will assume that irony is a super class which contains sarcasm. For us sarcasm is a specialisation of irony which implies the intention of harm or insult. This line is thin and it might be confusing even for native speakers since these concepts tend to be interchangeable; but by assuming sarcasm

* Authors thank Red Temática en Tecnologías del Lenguaje CONACyT for the support provided to the first author during part of this research.

as a type of irony and focusing on the irony phenomenon we warrant to include all ironic expressions. However we will not be able to distinguish sarcasm from irony. We consider satire a special use of irony in the context of humor, and since our approach aims to capture irony independent of the context, satire is out of the scope of this work.

The automatic detection of irony and sarcasm has been extensively studied in recent years in different languages. Studies have focused mainly in English (Reyes, Rosso, and Veale, 2013; González-Ibáñez, Muresan, and Wacholder, 2011; Barbieri and Saggion, 2014; Davidov, Tsur, and Rappoport, 2010; Tsur and Davidov, 2010). However other languages are also being studied: Chinese (Tang and Chen, 2014), Czech (Ptáček, Habernal, and Hong, 2014), Brazilian Portuguese (Vanin et al., 2013), Dutch (Liebrecht, Kunneman, and van den Bosch, 2013), Italian (Bosco, Patti, and Bolioli, 2013) and Portuguese (Carvalho et al., 2009). However, to our knowledge there has not been a study of the phenomenon in Spanish. In this work we look to establish the state of the art baselines for irony detection in Spanish.

Recent advances in detection of irony have shown that the supervised classification methodology with a great extent of feature engineering produces satisfactory indicators for irony or sarcasm. This methodology has been tested in short text such as product reviews, news commentaries and *tweets*. In this work we concentrate on producing classification baselines focused only on straightforward word and character features (i.e., n -grams) for posts from the social network *Twitter*. This is motivated by the promising results obtained in previous research for other languages. However, past approaches use a contrastive approach in which they look to differentiate two or more competing classes such as *humour* and *education*. Instead, we propose a binary classification between irony and non-irony, we consider such a classifier would be less domain/class dependent.

2 Related work

Recently there has been a surge in the study of irony and sarcasm detection for short texts. In their collaboration Mihalcea, Strapparava and Pulman proposed a system that identifies humorous one-liners, clas-

sified with Naive Bayes and Support Vector Machines (Mihalcea and Strapparava, 2006; Mihalcea and Pulman, 2007). Carvalho et al., introduced clues for automatically detecting irony in user generated content -user comments- in Portuguese (Carvalho et al., 2009). They distinguished from ironic, non-ironic, ambiguous and doubtful comments. Among their most satisfactory features were special punctuation, quotation marks and expressions of laughter. Tsur and Davidov built a system to recognise sarcastic sentences by analyzing patterns in sarcastic product reviews and using them to classify afterwards with k -Nearest Neighbors (Tsur and Davidov, 2010). To extract the ironic reviews, they relied on the star-based score of each review and compared it to the overall polarity of the review text. When the polarity did not match the star rating, an ironic instance was assumed.

Similar approaches were used for short texts extracted from the social network *Twitter*. Davidov et al., followed Tsur and Davidov’s baseline to recognise sarcasm in posts in English from Twitter (Davidov, Tsur, and Rappoport, 2010). This approach benefits from the user assigned tags called *hashtags* to automatically retrieve posts tagged as *#sarcasm* as the sarcastic class. Also working with a Twitter corpus in English, González-Ibáñez et al., used a series of lexical and pragmatic factors to identify sarcasm from positive (tagged by positive words, such as *#happy*) and negative (tagged by negative words, such as *#sad*) posts (González-Ibáñez, Muresan, and Wacholder, 2011). They used Logistic Regression and Support Vector Machines as classifiers. Liebrecht et al., (Liebrecht, Kunneman, and van den Bosch, 2013) automatically retrieved Dutch tweets tagged with *#sarcasme*, and classified sarcastic and non-sarcastic posts with a Balanced Winnow Algorithm. They employed stylistic features, such as word n -grams and punctuation marks, as well as intensifiers for Dutch and words that contained or derived from the word *sarcasm*. Reyes et al., worked with tweets in English as well (Reyes, Rosso, and Veale, 2013). This work crafted a multidimensional system based on signatures, unexpectedness, style and emotional scenarios to identify irony from politics, humor and education. Posts for all four classes were retrieved by extracting posts tagged

with *#irony*, *#politics*, *#humor* and *#education*, respectively. Barbieri et al., (Barbieri and Saggion, 2014) used the corpus built by Reyes et al., for their own approach. They also designed a set of features: frequency, written-spoken style, intensity, structure, sentiments, synonyms and ambiguity. They use an SVM classifier among the same classes as Reyes et al., Tungthamthiti et al., (Tungthamthiti, Kiyooki, and Mohd, 2014) devised a system which considers sentiment analysis, common-sense knowledge and coherence. They achieved generally favorable results, also using SVM as their classifier. Ptáček et al., proposed baselines for sarcasm detection for English and Czech with SVM and MaxEnt, obtaining the highest results – with stylistic n -gram based features – for English, and less satisfactory results for Czech on a manually annotated corpus of tweets (Ptáček, Habernal, and Hong, 2014).

Most of the above works experimented over a balanced corpus. That is, they trained and evaluated with equal number of samples per class. Noteworthy exceptions are Liebrecht et al., (2013) who tested with a realistic sample (in which sarcastic tweets account for less than 10% of the total), and self-designed distributions; such as Ptáček et al., (2014), who trained and tested with a proposed distribution of 25% ironic, 75% non-ironic experiment, and Reyes et al., (2013) with 30% and 70% respectively.

Many approaches have been formulated, along with features based on an interpretation of sarcasm and irony. Table 1 summarises the features for the works focused on *tweets*. For example, Reyes et al., (2013) used polarity skip-grams from the intuition that one generally employs positive terms to communicate a negative meaning when using irony. However, most authors report stylistic features as the better indicators for irony, whilst intuition-based features do not significantly contribute to their systems (Carvalho et al., 2009; Reyes, Rosso, and Veale, 2013). Intuition-based features tend to rely heavily on domain-specific markers and indicators that work well on fixed scopes that are not prone to change. This is observed in the way authors create a different set of features per language and domain.

It catches our attention that in English, Portuguese and Czech stylistic features such as word and character n -grams -as well as

punctuation marks and skip-grams- tend to be constantly meaningful.

3 Corpus generation

In the social networking website *Twitter*, people post messages of up to 140 characters, which are called *tweets*. These can contain references to other users, noted by *@user*; as well as tags to identify topics or words of interest, called *hashtags* and noted by a pound sign (e.g., *#thisIsAHashtag*). As done by previous work we use the manual tagging done by users of *Twitter* to recollect a corpus of ironic expressions.

3.1 Extraction and annotation

For this paper, we required a large set of ironic and non-ironic *tweets* in Spanish. We follow the general extraction method of Reyes et al., (2013) and Liebrecht et al., (2013), where they rely on user tags and assume they are correct. *Tweets* tagged as *#irony* are considered ironic without further verification. Note that an unbiased view of the use of these tags in Twitter can point to what the majority of users consider to be irony, and not necessarily to a formal definition of it.

As stated above, we assume an interpretation of irony that encapsulates sarcasm as a subclass of it, and consider sarcastic tweets to be ironic. A manual inspection of tweets tagged as *#ironía* and *#sarcasmo* (irony and sarcasm, respectively) shows that the tags are often used interchangeably. It is likely that some subset of users of *Twitter* cannot tell the difference themselves. This is understandable since the line between the two concepts is thin and people in social media are not interested in the strict definition of what they write but the intention. Following this consideration, we extract tweets tagged as both irony and sarcasm for the ironic set of our corpus¹.

For the ironic part of the corpus, we turn to tweets annotated by users as *#ironía* and *#sarcasmo*, searching only results in Spanish for irony and sarcasm. We collect the non-ironic *tweets* using empty words as search terms (*quién*, *cómo*, *cuándo*, *dónde*, *por qué*, which translates to who, how, when, where, why; among others found in table ??), avoiding tweets tagged as ironic. That is, any tweet that is not explicitly tagged as ironic

¹Tweets were collected through the Twitter API for the Ruby programming language

Carvalho et al., (2009)	
Stylistic	punctuation marks, quotation/ exclamation/ question marks, laughter expressions, diminutives
Linguistic	interjection words, demonstrative determiners, named entity recognition
Emotional	requires presence of positive adjectives/nouns not surrounded by negatives
Davidov et al., (2010)	
Stylistic	punctuation marks, sarcastic patterns learned by SASI
González-Ibáñez et al., (2011)	
Stylistic	word unigrams, emoticons, user mentions
Linguistic	linguistic processes (adjectives, pronouns, etc.)
Emotional	psychological processes (positive, negative), affective WordNet
Liebrecht et al., (2013)	
Stylistic	word unigrams, bigrams and trigrams, exclamation marks
Linguistic	Dutch intensifiers, marker words derived from <i>sarcasm</i>
Reyes et al., (2013)	
Stylistic	punctuation marks, c-grams, skip-grams
Linguistic	various verb, temporal adverb, opposing term and semantic field counts
Emotional	polarity s-grams, dictionaries for activation, imagery, pleasantness
Barbieri et al., (2014)	
Stylistic	punctuation, word length, emoticons
Linguistic	POS-tagger count by label, common vs rare synonym use
Emotional	gap between rare and common words
Emotional	word intensity, gap between positive and negative terms
Tungthamthiti et al., (2014)	
Stylistic	punctuation, word unigrams, bigrams and trigrams, emoticons, slang words
Linguistic	grammatical coherence of the sentence is quantified
Emotional	word polarity, sentiment contradiction

Table 1: Features by author and type

Words	Translation
donde / where	dónde / where (q.)
quien / who	quién / who (q.)
como / as	cómo / how (q.)
cuando / when	cuándo / when (q.)
este / this	esta / this
tiene / has	está / is
porque / because	por qué / why

Table 2: Words searched to recover the non-ironic dataset (q.:question)

is considered non-ironic. This is based in the work of Liebrecht et al., (2013), where the non-sarcastic class is called “background”. We consider this to be less biased towards a certain domain or class. Figure 1 illustrates this and the different approaches.

3.2 Normalization

In order to normalize the corpus duplicate tweets are automatically removed. Our corpus contains approximately 14,500 unique ironic tweets and 670,000 unique non-ironic tweets. Table ?? summarises the main characteristic of the corpus. Additionally, we nor-

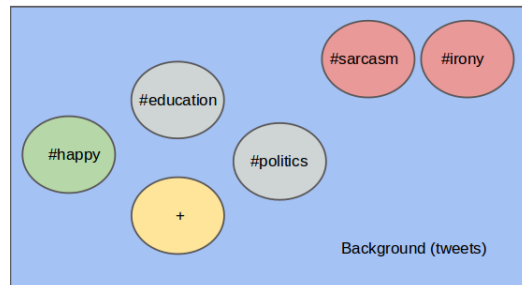


Figure 1: Various approaches to non-ironic classes: González-Ibáñez et al., (2011) positive/negative/sarcastic classes, Reyes et al., (2013) irony/politics/humor/education classes, and Liebrecht et al., (2013) sarcasm/non-sarcasm classes.

malize all hyperlinks and user references under one symbol each (*http://link* and @, respectively) so that an algorithm can consider all of the different user references as a single semantic element, *referencing a user*, that happens in many tweets and might relate to irony or non-irony, with a similar reasoning for hyperlinks.

We don’t edit punctuation or non-unicode characters, but we do get rid of excessive spaces, lowercase all text, and tokenize punc-

	Non-ironic	Ironic
Tweets	675,671	14,511
Tokens	11,168,897	219,964
Types	346,967	28,033
Diversity	0.03	0.12
Avg. length	3	4

Table 3: Characteristic of corpus: size and lexical diversity

tuation marks as different words (separated by spaces).

Hashtags that comprise or include the words *irony* and *sarcasm* are also removed, but the rest of *hashtags* are left without the # symbol. That is, all other *hashtags* are plain text as the rest of the tweet. This decision is based on a current tendency to express deeper meaning on *hashtags* than just tagging a topic. For instance, some tweets are formed by only a hyperlink and a set of *hashtags* in which all the message is communicated. Removing the *hashtags* completely may produce an empty message where meaning was intended.

As an example of this normalization the following *tweets* *Who will be the champ in #LigaMx #sarcasm* and *You never play for free, never @EliRom #sarcasm http://t.co/bjRrKv7kpZ* become *Who will be the champ in LigaMx* and *You never play for free, never @ http://link* respectively.

An additional set of 1,483 ironic *tweets* was collected. This set is used as a testing corpus.

4 Our approaches

We tested two levels for irony detection for tweets in Spanish: word and character based.

4.1 Word based

At this level, we implemented two word-based baselines: The first one is a collection of word *n*-grams. Following previous studied approaches on irony classification we use a sparse representation of the *tweets* to train a SVM and a Random Forest classifier. This sparse representation is formed by typical unigram, bigram and trigram counts.

The second baseline uses a distributed representation of the tweets based on the *word2vec* approach (Mikolov et al., 2013; Le and Mikolov, 2014). *Word2vec* is a two-layer neural network that processes an unlabeled input corpus and outputs a series of word

vectors. *Word2vec* groups vectors of semantically similar words in a vector space, in which distances between them can be measured. This distance among words depends on the context in which they are used. Given enough data, *word2vec* has proved to make precise assumptions about the meaning of a word based on past occurrences. These can be used to establish the relationship of a word with its context by means of vector operations.

4.2 Character based

For the character based approach we use character *n*-grams, a feature that proved to be representative in the works of Reyes et al., (2013) and Ptáček et al., (2014). To figure which *n*-grams to use, we measured the average word size for both sets in the corpus. It was roughly 4 for both, and as to consider whole words too, we decided on character bigrams, trigrams and tetragrams. This feature is also able to collect other relevant characteristics in the literature, such as punctuation marks and *emoji*.

User generated content is plagued with erratic writing, spelling mistakes being a popular example. Character *n*-grams have the advantage of adapting to users' vices by identifying *n*-grams that account for a certain linguistic trait, such as a lemma, a prefix or a suffix of some words. For example, the following four spellings for the word *este* (*this*, in Spanish) were found *este*, *estee*, *eeeste*, *eestee*. All of these contain the trigram *est*, even if three of them are not spelled correctly. With word based approaches, this kind of diversity results in many features with low frequency.

4.3 Implementation

Our experimentation was performed using Support Vector Machines and Random Forests classifiers. For both, we used the *scikit-learn* implementation². SVM has a lineal kernel and its configuration was not changed. In the case of Random Forests we used 1,000 estimators. The decision to use these classifiers is driven by previous works: Ptáček et al., (2014) and González- Ibáñez et al., (2011) use SVM; and Reyes et al., (2013) use Decision Trees, which we replace with Random Forests.

²<http://scikit-learn.org/stable/>

$tf-idf$ (term frequency-inverse document frequency) is used for word/character representation, as it tends to favor relevant terms among all documents. Common empty words are excluded with a list of stop words. Words with very low $tf-idf$ score are also excluded. To create the distributional model we train a vector space using *doc2vec*³ with approximately 660,000 non ironic tweets implementing a c-bow model. We take the necessary measures to ensure that the distributional model did not contain testing non-ironic *tweets* depending on the evaluation setting.

5 Experiments and results

For our evaluation we use different versions of the dataset. We use the standard balanced dataset setting, in which there are equal elements per class. Additionally, we use Reyes et al., (2013) proposed unbalanced set with 70% non-ironic and 30% ironic tweets. Furthermore, we propose a third distribution, 90% non-ironic and 10% ironic tweets, which we believe to be more realistic. Table 4 shows the number of tweets used in each case. Besides changing the proportions on the dataset, we tested three baselines: *word-gram* is based on representing the tweet as a sparse vector of the word $td-idf$ weights; *word2vec* is based on representing the tweet as a distributional vector based on *word2vec*; finally, *char-gram* is based on representing the tweet as a sparse vector of the character $td-idf$ weights. In the following subsections we present the main results for each built baseline.

	Ironic	Non-ironic
	Balanced	
Train	14,511	14,511
Test	1,483	1,483
	Unbalanced	
Train	14,511	33,859
Test	1,483	3,458
	Proposed	
Train	14,511	130,599
Test	1,483	13,347

Table 4: Dataset distributions used in this work

5.1 Balanced

Table 5 presents the results for each baseline on the balanced setting. It also sum-

³From the Gensim library: <https://radimrehurek.com/gensim/models/doc2vec.html>

Baseline	RF	SVM
Word level		
<i>word-gram</i>	0.68	0.67
<i>word2vec</i>	0.76	0.78
For other languages		
Davidov et al., (2010)	N/A	0.83
Tungthamthiti et al., (2014)	N/A	0.79
Character level		
<i>char-gram</i>	0.87	0.86
For other languages		
Reyes et al., (2013)	N/A	0.71
Ptáček et al., (2014)(ENG)	N/A	0.93

Table 5: F-Measures for all baselines under balanced distributions (50-50) and *tweets* datasets

marizes previous performances in other languages when comparable. We notice that our baseline systems for Spanish are comparable with the previous work in other languages. At the word level, we observe that *word2vec* surpasses *word-gram* baseline. Understandably, since *word2vec* word vectors consider an extensive depiction of *Twitter* language in order to calculate the distributional model. Our best result at this level, 0.78 f-score with SVM, is closest to Tungthamthiti et al., (2014) which was of 0.79 with a balanced dataset for the English language and a larger set of features. On the other hand, we notice that the best performance is at the character level, **0.87** with a Random Forests classifier. At this level it is second only to the best ever result by Ptáček et al., (2014), higher than the score for English by Reyes et al., (2010), and widely better than previous attempts at Czech and Portuguese. In conclusion, we believe this is a comparable baseline to previous work done in other languages in a balanced setting.

5.2 Unbalanced

Table 6 presents the results for each baseline on the unbalanced setting. We immediately notice that the performance considerably fell. This was something we expected, however the severity of the fall at the word level was unforeseen. On the other hand, at the character level the baseline fall was not equally harsh, 0.80 for both types of classifiers. A closer inspection into the results shows that classifying with Random Forests has a class F-Measure of 0.80 for the irony class and 0.93 for the non-irony class, while for SVM is 0.80 and 0.92. Reyes et al., (2013) and Ptáček

Baseline	RF	SVM
Word level		
<i>word-gram</i>	0.48	0.37
<i>word2vec</i>	0.38	0.61
For other languages		
Reyes et al., (2013)	N/A	0.53
Ptáček et al., (2014) (<i>ENG 1:3</i>)	N/A	0.92
Character level		
<i>char-gram</i>	0.80	0.80

Table 6: F-Measures for all baselines under unbalanced distributions (30-70) and *tweets*

et al., (2014) propose similar unbalanced settings (30-70 and 25-75 respectively), the first one reporting a severe drop in the performance.

5.3 New proposed distribution

Given the outcome with the unbalanced distribution, we wanted to test the resilience of the character level representation with a more realistic distribution. For this we proposed a setting with 10% ironic, and 90% non-ironic elements. Table 7 presents the results for the character level baseline in this new distribution. For both classifiers the performance declines, however in the case of SVM the performance continues being competitive at a 0.74 F-score.

Baseline	RF	SVM
<i>char-gram</i>	0.61	0.74

Table 7: F-Measures for char-gram baseline under proposed distribution (90-10)

5.4 Discussion

Our best scores came from from the character level baseline for all three settings of the corpus, indicating that character n-grams are good indicators for irony in Spanish. It is possible that for very short texts such as *tweets*, word based features fail to assimilate enough information per *tweet* to represent it correctly, whereas a character based model will split a sentence into many more features, having a clearer picture of each *tweet*.

After these results, we explored the counts and *tf-idf* weights of the most common features and found that *emoji* and *smileys* have very high scores. Expressions of laughter such as *jajaja* and *lol* exist in both ironic and non-ironic datasets, but are more representative of the ironic, in accordance to Carvalho

et al., (2009), Reyes et al., (2013), Davidov et al., (2010), and Liebrecht et al., (2013).

We also observe a high count for common morphemes in Spanish. We theorize that character *n*-grams also have high morphological information to offer, and are able to collect common morphemes to use as low or high value features for a certain class.

6 Conclusions

We proposed a binary classification approach for ironic and non-ironic short texts. For such purpose we focus on verbal irony and our concept of irony includes sarcasm. Following previous proposals we take advantage of manually tagged *tweets* (i.e., *#ironía* and *#sarcasmo*). We produced three classification baseline systems focused only on straight-forward word and character features for *tweets* in Spanish. The first baseline consisted on representing the *tweets* as *tf-idf* weights from the word *n*-grams. The second consisted on a distributional representation of the *tweets*. Finally, the third baseline represented *tweets* as *tf-idf* weights from the character *n*-grams.

Our approaches reached comparable results to related works in other languages. This points out to the validity of our baselines. However, during our experimentation we identify that the character level baseline outperformed other approaches for other languages. We achieved F-Measures of 0.87 on a balanced dataset using a Random Forest classifier, 0.80 on an unbalanced setting (70/30) and 0.74 on an even more unbalanced but more realistic dataset (90/10), in both cases using an SVM classifier.

We observed that character-based features are good indicators for irony detection, and generally offer a good baseline for Spanish. We believe that by providing a solid baseline that delivers acceptable performance, researchers can focus on developing domain-specific features which we did not incorporate in this work and improve these results. Further studies could focus on the use of linguistic based features in order to better characterise irony or try to distinguish irony from sarcasm. Additionally, as a part of the research we collected a large set of ironic and non-ironic *tweets*. Such collection is an open resource for further use by the research community⁴.

⁴The resource can be downloaded from here:

References

- Barbieri, Francesco and Horacio Saggion. 2014. Modelling irony in twitter: Feature analysis and evaluation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 26–31, may.
- Bosco, Cristina, Viviana Patti, and Andrea Bolioli. 2013. Developing corpora for sentiment analysis: The case of irony and senti-tut. *IEEE Intelligent Systems*, (2):55–63.
- Carvalho, Paula, Luís Sarmiento, 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.
- Davidov, Dmitry, Oren Tsur, and Ari Rappoport. 2010. Semi-Supervised recognition of sarcastic sentences in twitter and amazon. In *Proceeding of the 23rd international conference on Computational Linguistics (COLING)*, July.
- González-Ibáñez, Roberto, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the ACL: HLT: short papers-Volume 2*, pages 581–586.
- Haverkate, Henk. 1990. A speech act analysis of irony. *Journal of Pragmatics*, 14(1):77–109.
- Le, Quoc V and Tomas Mikolov. 2014. Distributed representations of sentences and documents. *arXiv preprint arXiv:1405.4053*.
- Liebrecht, C., F. Kunneman, and A. 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*.
- Mihalcea, Rada and Stephen Pulman. 2007. Characterizing humour: An exploration of features in humorous texts. In *Computational Linguistics and Intelligent Text Processing*. Springer, pages 337–347.
- Mihalcea, Rada and Carlo Strapparava. 2006. Learning to laugh (automatically): Computational models for humor recognition. *Computational Intelligence*, 22(2):126–142.
- Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- Ptáček, Tomáš, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on czech and english twitter. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 213–223.
- Reyes, Antonio, Paolo Rosso, and Davide Buscaldi. 2012. From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering*, 74:1–12.
- Reyes, Antonio, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation*, 47(1):239–268.
- Tang, Yi-jie and Hsin-Hsi Chen. 2014. Chinese irony corpus construction and ironic structure analysis. pages 1269–1278.
- Tsur, Oren and Dmitry Davidov. 2010. Icwsm - a great catchy name: Semi-supervised recognition of sarcastic sentences in product reviews. In *International AAAI Conference on Weblogs and Social*.
- Tungthamthiti, Piyoros, Shirai Kiyooki, and Masnizah Mohd. 2014. Recognition of sarcasms in tweets based on concept level sentiment analysis and supervised learning approaches. In *Proceedings of the 28th Pacific Asia Conference on Language, Information, and Computation*.
- Vanin, Aline A, Larissa A Freitas, Renata Vieira, and Marco Bochernisan. 2013. Some clues on irony detection in tweets. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 635–636. International World Wide Web Conferences Steering Committee.