Do We Criticise (and Laugh) in the Same Way? Automatic Detection of Multi-Lingual Satirical News in Twitter [*]

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

Abstract

During the last few years, the investigation of methodologies to automatically detect and characterise the figurative traits of textual contents has attracted a growing interest. Indeed, the capability to correctly deal with figurative language and more specifically with satire is fundamental to build robust approaches in several sub-fields of Artificial Intelligence including Sentiment Analysis and Affective Computing.

In this paper we investigate the automatic detection of Tweets that advertise satirical news in English, Spanish and Italian. To this purpose we present a system that models Tweets from different languages by a set of language independent features that describe lexical, semantic and usage-related properties of the words of each Tweet. We approach the satire identification problem as binary classification of Tweets as satirical or not satirical messages. We test the performance of our system by performing experiments of both monolingual and cross-language classifications, evaluating the satire detection effectiveness of our features. Our system outperforms a word-based baseline and it is able to recognise if a news in Twitter is satirical or not with good accuracy. Moreover, we analyse the behaviour of the system across the different languages, obtaining interesting results.

1 Introduction

Satire is a form of language where humour and irony are employed to criticise and ridicule someone or something. Even if often misunderstood, "in itself, satire is not a comic device —it is a critique — but it uses comedic devices such as parody, exaggeration, slapstick, etc. to get its laughs." [Colletta, 2009]. Satire is distinguished by figurative language and creative analogies, where the fiction pretends to be real. Satire is also characterised by emotions (like anger and disappointment) that are hard to detect due to their ironic dimension.

The ability to properly detect and deal with satire would be strongly beneficial to several fields where a deep understanding of the metaphorical traits of language is essential, including Affective Computing [Picard, 1997] and Sentiment Analysis [Turney, 2002; Pang and Lee, 2008]. Looking at the big picture, computational approaches to satire are fundamental to build a smooth human-computer interaction, improving the way computers interpret and respond to peculiar human emotional states.

In this paper we study the characterisation of satire in social networks, experimenting new approaches to detect satiric Tweets inside and across languages. We retrieve satirical Tweets from popular satirical news Twitter accounts, in English, Spanish and Italian. We rely on these accounts since their content is a contribution of several people and their popularity reflects the interest and appreciation for this type of language. We compare the Tweets of satirical news accounts with Tweets of popular newspapers, advertising actual news. A few examples from our dataset are the following ones:

- **Satirical News**
  English: Police Creative Writing Awards praise 'most imaginative witness statements ever'.
  Spanish: Artur Mas sigue esperando el doble "check" de Mariano Rajoy tras la votación del 9-N.
  (Artur Mas is still waiting for Mariano Rajoy's double check after 9-N consultation).
  Italian: "Potrei non opporre veti a un presidente del Pd", ha detto Berlusconi iscrivendosi al Pd.
  ("I might not limit powers of Democratic Party president", said Berlusconi enrolling in the Democratic Party).

- **Non-Satirical News**
  English: Major honours for The Times in the 2014 British Journalism Awards at a ceremony in London last night.
  Spanish: Rajoy admite que no ha hablado con Mas desde que se convocó el 9-N y que no sabe quién manda ahora en Cataluña.
  (Rajoy admits that he hasn't talked to Mas since the convocation of 9-N consultation and that he doesn't know who's governing in Catalonia).
  Italian: Berlusconi e il Colle: "Non metterò veti a un

[*] The research described in this paper is partially funded by the Spanish fellowship RYC-2009-04291 and the SKATER-TALN_UPF project (TIN2012-38584-C06-03).

candidato Pd".

(Berlusconi states: "I will not limit powers of Democratic Party president").

In these examples we can see that satire is used to criticise and convey a peculiar hidden meaning to the reader. The satirical English example is a critic against police and its dishonest way of solving issues by "inventing" witnesses. The satirical Spanish Tweet is a critic against Rajoy (Prime Minister of Spain at the time of writing), as he did not want to discuss with Mas (Prime Minister of Catalonia at the time of writing) the decision of doing a consultation on November 9th (on the Catalonia independence). For this reason "Mas is still waiting for him to consider it". The satirical Tweet in Italian criticises the power that Berlusconi had in Italy even though he was not Italian prime minister any more.

Our system relies on language-independent *intrinsic word features* (word usage frequency in a reference corpus, number of associated meanings, etc.) and on language dependent *word-based features* (lemmas, bigram, skip-gram). As classifier we employ the supervised algorithm Support Vector Machine[1] [Platt, 1999] because it has proven effective in text classification tasks.

The contributions of this paper are: (1) a novel language-independent framework to detect satire in Twitter, (2) a set of experiments to test our framework with English, Spanish and Italian Tweets, (3) a set of cross-language experiments to analyse similarities and differences in the use of satire in English, Spanish, and Italian, and (4) a dataset composed of satirical news Tweets and non-satirical news Tweets in English, Spanish and Italian.

Our paper includes seven sections. In the second section we provide an overview of the state of the art on satire and related AI topics. In Section 3 we describe the tools we used to process Tweets in the three languages. In Section 4 we introduce the features we exploit to detect satiric Tweets in different languages, and in Section 5 we introduce and evaluate the experiment we carried out to test the performances of our satire-detection system. In the last two sections we discuss the cross-language abilities of our system, showing its behaviour across different languages. We then summarise our work in the last Section.

2 Literature Review

Satire is a form of communication where humour and irony are used to criticise someone's behaviour and ridicule it. Satirical authors may be aggressive and offensive, but they "always have a deeper meaning and a social signification beyond that of the humour"[Colletta, 2009]. Satire loses its significance when the audience does not understand the real intents hidden in the ironic dimension. Indeed, the key message of a satirical utterances lays in the figurative interpretation of the ironic sentence. Satire has been often studied in literature [Peter, 1956; Mann, 1973; Knight, 2004; LaMarre *et al.*, 2009], but rarely with a computational approach. The work of Burfoot and Baldwin [2009] attempts to computationally model satire in English. They retrieved news-wires documents and satire news articles from the web, and build a model able to recognise satirical articles. Their approach included standard text classification, lexical features (including profanity and slang) and semantic validity where they identify the named entities in a given document and query the web for the conjunction of those entities.

The use of irony in satire is fundamental. The traditional definition of irony is "saying the opposite of what you mean" [Quintilien and Butler, 1953]. Since 2010 researchers designed models to detect irony automatically. Veale [2010] 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 skip-grams which capture word sequences that contain (or skip over) arbitrary gaps, are the most informative features. Barbieri and Saggion [2014a] and [2014b] designed an irony detection system that avoided the use of the word-based features. However, irony has not been study intensively in languages other than English. A few researches has been carried out on irony detection on other languages like Portuguese [Carvalho *et al.*, 2009; de Freitas *et al.*, 2014], Dutch [Liebrecht *et al.*, 2013] and Italian [Barbieri *et al.*, 2014].

Affective computing (AC) is a well known AI field dealing with Human-Computer Interaction. Affective computing studies intelligent systems that are able to recognise, process and generate human emotions. Emotions are relevant for AI as they "play roles not only in human creativity but also in rational human thinking and decision making as computers that will interact naturally and intelligently with humans need the ability to at least recognise and express affect" [Picard, 1997]. There are many AC applications [Tao and Tan, 2005] including computer vision (emotion of a face and body language), wearable computing, and all the natural language processing area (from the content to the voice tone).

3 Text Analysis and Tools

We associated to each Tweet a normalised version of its text by expanding abbreviations and slang expressions, properly converting hashtags into words whether they have a syntactic role (i.e. they are part of the sentence), and removing links and mentions ("@twitter-user"). We describe in this section the tools and dataset we used.

3.1 English Tools

We made use of the GATE application TwitIE [Bontcheva *et al.*, 2013] where we enriched the normaliser, adding new abbreviations, new slang words, and improving the normalisation rules. We also employed TwitIE for tokenisation, Part Of Speech (POS) tagging and lemmatisation. We used WordNet [Miller, 1995] to extract synonyms and synsets of a word. We employed the sentiment lexicon SentiWordNet3.0 [Baccianella *et al.*, 2010]. Finally, the American National Corpus [2] has been employed as frequency corpus to obtain the usage frequency of words in English.

[1] LibLINEAR: http://www.csie.ntu.edu.tw/cjlin/liblinear

[2] http://www.anc.org/

Language	Non-Satirical	Satirical
English	The Daily Mail (UK) The Times (UK)	NewsBiscuit The Daily Mash
Spanish	El Pais El Mundo	El Mundo Today El Jueves
Italian	Repubblica Corriere della Sera	Spinoza Lercio

Table 1: List of Twitter accounts of newspaper and satirical news in British English, Iberian Spanish, and Italian.

3.2 Spanish Tools

We relied on the tool Freeling [Carreras *et al.*, 2004] to perform sentence splitting, tokenisation, stop words removal, POS tagging, and Word Sense Disambiguation. WSD in Freeling using the Spanish Wordnet of the TALP Research Centre, mapped by means of the Inter-Lingual-Index to the English Wordnet 3.0 whose synset IDs are in turn characterised by sentiment scores by means of SentiWordnet. As corpus frequency we used the texts of a dump of the Spanish Wikipedia as of May 2014.

3.3 Italian Tools

We tokenised, POS tagged, applied Word Sense Disambiguation (UKB) and removed stop words from the normalised text of Tweets by exploiting Freeling. We also used the Italian WordNet 1.6[3] to get synsets and synonyms of each word of a Tweet as well as the sentiment lexicon Sentix [Basile and Nissim, 2013] derived from SentiWordnet to get the polarity of synsets. We relied on the CoLFIS Corpus frequency of Written Italian[4].

3.4 Dataset

In order to train and test our system we retrieved Tweets from twelve twitter accounts from June 2014 to January 2014. We considered four Twitter accounts for each language (English, Spanish and Italian), and within each language two are satirical and two are non-satirical newspapers. They are shown in Table 1.

After downloading the Tweets we filtered them removing the Tweets that were not relevant to our study (for instance: "Buy our t-shirt" or "Watch the video"). We left only Tweets that were actual news (satirical or non-satirical). In order to have a balanced dataset, with the same contribution from each Twitter account, we selected 2,766 Tweets randomly from each account, obtaining a total of 33,192 Tweets, where half (16,596) were satirical and half were non-satirical news (2,766 was the least number of Tweets that a single account included, which was the Italian satirical account "Lercio"). We shared[5] this dataset as a list of Tweet IDs since per Twitter policy it is not possible to share the text of the Tweets.

4 Our Model

We characterised each Tweet by six classes of features: (1) Word-Based, (2) Frequency, (3) Synonyms, (4) Ambiguity, (5) Part of Speech, (6) Sentiments, and (7) Punctuation. Some of these features were aimed at capturing common word-patterns (1) and others to describe intrinsic aspects of the words included in each Tweet (2-6). The interesting propriety of the intrinsic word features is that they do not rely on words-patterns, hence can be used across languages.

4.1 Word-Based

We designed this group of features to build our baseline, since word based features are usually very competitive in text classification tasks. We computed the five word-based features: *lemma* (lemmas of the Tweet), *bigrams* (combination of two lemmas in a sequence) and *skip 1/2/3 gram*.

4.2 Frequency

We accessed the frequency corpora (of each language, Section 3) to retrieve the frequency of each word of a 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 including all the words of each Tweet. We also determined these features by considering only Nouns, Verbs, Adjectives, and Adverbs. Moreover, we count the number of bad/slang words in the Tweet (using three lists we compiled for each language). The final number of Frequency features is 16.

4.3 Ambiguity

To model the ambiguity of the words in the Tweets we use the WordNet synsets associated to each word. Our hypothesis is that if a word includes several 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 *mean synset number* of all the words, and the *synset gap* that is the difference between the two previous features. We determine the value of these features by including all the words of a Tweet as well as by considering only Nouns, Verbs, Adjectives or Adverbs separately. The Ambiguity features are 15.

4.4 Part Of Speech

The features included in the Part Of Speech (POS) group are designed to capture the syntactic structure of the Tweets. The features of this group are eight and each one 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 *Appositions*.

4.5 Synonyms

We consider the frequencies (for each language its own frequency corpora, see Section 3) of the synonyms of each word in the Tweet, as retrieved from WordNet. Then we computed, across all the words of the Tweet: the *greatest* and the *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

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

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

[5] http://sempub.taln.upf.edu/tw/ijcai2015/

present in the Tweet. We determine also the greatest/lowest number of synonyms and the mean number of synonyms of the words with frequency greater/lower than the one present in the Tweet (*gap* feature). We computed the set of Synonyms features by considering both all words of the Tweet together and only words belonging to each one of the four Parts of Speech listed before.

4.6 Sentiments

The sentiments of the words in Tweets are important for two reasons: to detect the *sentiment* (e.g. if Tweets contain mainly positive or negative terms) and to capture unexpectedness created by a negative word in a positive context or vice versa. Relying on the three Sentiment lexicons described in Section 3, we computed the *number of positive / negative words*, the *sum of the intensities of the positive / negative scores of words*, the *mean of positive / negative score of words*, the *greatest positive / negative score*, the *gap between the greatest positive / negative score and the positive / negative mean*. Moreover we simply count (and measure the ratio) the *words with polarity* not equal to zero, to detect subjectivity in the Tweet. As previously done, we computed these features by considering only Nouns, Verbs, Adjectives, and Adverbs.

4.7 Characters

These features were designed to capture the punctuation style of the satirical Tweets. Each feature that is part of this set is the number of a specific punctuation mark, including: ".", "!", "?", "$", "%", "&", "+", "-", "=". We also compute numbers of Uppercase and Lowercase characters, and length of the Tweet.

5 Experiments and Results

In order to test the performances of our system we run monolingual and cross-lingual experiments.

5.1 Monolingual Experiments

In order to test the performances of our system we run two kind of balanced binary classification experiments, where the two classes are "satire" and "non-satire". We gathered three datasets of English, Spanish and Italian Tweets; each dataset includes two newspaper accounts, N1 and N2, and two satirical news accounts, S1 and S2.

In the **first binary balanced classification experiment**, we train the system on a dataset composed of 80% of Tweets from one of the newspaper accounts and 80% of Tweets from one of the satirical accounts (5,444 Tweets in total). Then we test the system on a dataset that includes 20% of the Tweets of a newspaper account that is different from the one used for training and 20% of the Tweets of a satirical account that has not been used for training. The final size of our testing set is 1,089 Tweets. We run the following configurations:

- Train: 80% N1 and 80% S1 / Test: 20% N2 and 20% S2
- Train: 80% N1 and 80% S2 / Test: 20% N2 and 20% S1
- Train: 80% N2 and 80% S1 / Test: 20% N1 and 20% S2
- Train: 80% N2 and 80% S2 / Test: 20% N1 and 20% S1

It is relevant to remark that thanks to these training and test set configurations, we never use Tweets from the same account in both the training and testing datasets, thus we can evaluate the ability of our system to detect satire independently from the linguistic and stylistic features of a specific Twitter account. As a consequence we avoid the *account modelling / recognition* effect, as the system is never trained on the same accounts where it is tested.

In the **second binary balanced classification experiment**, the training set is composed of all the Tweets of each account. The dataset include 33,192 Tweets, and we evaluate the performance of our SVM classifier by a 5-folds cross validation.

For each experiment we evaluate a word-based model (W-B, word-based features from Section 4.1) that we consider our baseline, a model that relies on intrinsic word features (described from Section 4.2 to Section 4.6), and a third model that includes all the features of Section 4.

English Experiments

In Table 2 are reported the results of the English classification. When training on The Times and The Daily Mash and testing on the others the word-based features obtained same results than our models (F1 of 0.632 versus 0.635). In all the other cases, including the classification of all satirical Tweets versus all non satirical ones (N1+N2 vs S1+S2), the intrinsic-word model outperforms the word-based one. We can note that the results of the second experiment are higher with respect to any feature set, especially if we consider word-based features. We have to highlight that unlike the first experiment, in the second experiment Tweets from the same accounts are used both for training and testing, thus the system is able to learn to recognise besides satire, also the language and writing style features of each account.

When we extended the intrinsic word feature set by adding also word based features, we can observe that the performances of our classifiers improved (up to 0.678 in one combination, and up to 0.801 in the union of the accounts). According to the information gain scores, the best features in the N1+N2 vs S1+S2 dataset (see Table 6) belongs to the groups Character (length of the Tweet, the number of First uppercase words), POS (number of nouns) and Sentiment groups (ratio of words with polarity), Ambiguity (synset gap of nouns) and Frequency (rarest word frequency).

Train	Test	W-B	Intrinsic	All
N1S1	N2S2	0.646	**0.736**	0.683
N1S2	N2S1	0.610	**0.621**	0.660
N2S1	N1S2	0.641	**0.659**	0.678
N2S2	N1S1	0.632	0.635	0.639
N1N2S1S2	*5-fold*	0.752	**0.763**	0.801

Table 2: English monolingual classification. The table shows the F1 of each model, where N1=The Daily Mail, N2=The Times, S1=NewsBiscuit and S2=The Daily Mash. In **bold** the best results (not by chance confirmed by two-matched-samples t-test with unknown variances) between word-based and Intrinsic models.

Spanish Experiments

The Spanish model performances are reported in Table 3. F-measures are promising, with the best score when training on the accounts El Mundo and El Mundo Today (0.805 using only intrinsic word features). The intrinsic word features outperformed the word-based baseline in all the classifications. When adding the word-based features to the intrinsic features the results decrease in three cases out of four. Moreover word-based model obtained worse results also in the N1+N2 vs S1+S2 classification, even with the chance of modelling specific accounts. We can see in Table 6 that best features for Spanish were the Character (length, uppercase character ratio), POS (number of noun and appositions) and Frequency group (frequency gap of nouns, rarest noun and rarest adjective) and Ambiguity (mean of the number of synsets).

Train	Test	W-B	Intrinsic	All
N1S1	N2S2	0.622	**0.754**	0.727
N1S2	N2S1	0.563	**0.712**	0.723
N2S1	N1S2	0.592	**0.805**	0.709
N2S2	N1S1	0.570	**0.778**	0.737
N1N2S1S2	*5-fold*	0.738	**0.816**	0.852

Table 3: Spanish monolingual classification. The table shows the F1 of each model, where N1=El Pais, N2=El Mundo, S1=El Mundo Today, and S2=El Jueves. In **bold** the best results between word-based and Intrinsic models (same statistical test than English).

Italian Experiments

In the Italian experiments (Table 4) the intrinsic-word model outperformed the word-based model in all the combinations obtaining the best result when training on Repubblica and Lercio and testing on the other accounts (F1 are respectively 0.746 and 0.541). Incorporating word-based features to the intrinsic-word features model increased the F1 in two cases and decrease in the other two. However in the second type of experiment adding word-features helps. In Table 6 we can see that the best groups of features to detect satire was Characters (uppercase and lowercase ratio, length) POS (number of verbs), Ambiguity (verb synset mean, gap and max number of synset), Frequency (verb mean, gap, and rarest). In general, verbs seems play an important role in satire detection in Italian.

Train	Test	W-B	Intrinsic	All
N1S1	N2S2	0.518	**0.725**	0.672
N1S2	N2S1	0.541	**0.746**	0.674
N2S1	N1S2	0.527	**0.618**	0.640
N2S2	N1S1	0.578	**0.612**	0.625
N1N2S1S2	*5-fold*	0.739	**0.800**	0.842

Table 4: Italian monolingual classification. The table shows the F1 of each model, where N1=Repubblica, N2=Corriere della Sera, S1=Spinoza, and S2=Lercio. In **bold** the best results between word-based and Intrinsic models (same statistical test than English).

5.2 Cross-Lingual Experiments

In addition to these experiments focused on a single language, we also analysed the performances of a system composed of only language independent features (intrinsic features, features form 2 to 6 in Section 4) in a multi-lingual context, running two types of experiments. In the **first cross-language experiment** we train our model on the Tweets in a language and test the model over the Tweets of a different language; in this way we can see if the satirical accounts of different languages cross-reinforce the ability to model satire. By considering each language pair, we trained our satirical Tweet classifier on a language and tested it on another one.

We carry out these experiments to gain a deeper understanding of our intrinsic model assessing whether a model induced from one language can be used to detect the satire phenomena in a different language.

The **second cross-language experiment** was a 5-folds cross validation over all the dataset, including the Tweets from all the accounts of all the languages (total of 22,228 Tweets, where 16,596 were satirical and 16,596 non-satirical news).

Table 5 shows the results of the cross-lingual experiments (F1 of Non-Satirical and Satirical classes and the mean). A model trained in one language is not always capable of recognising satire in a different language. For example, a model trained in Italian is not able to recognise English and Spanish satire (F1 of 0.05 and 0.156). However, when testing in Italian and training in English and Spanish the system obtains the highest F1 scores of this type of experiment (respectively 0.632 and 0.695). When testing in English the system recognises satire (0.669) but not newspapers (0.031) when trained in Spanish, and vice versa when trained in Italian (good F1 for non-satirical newspaper, but low for satire). When testing Spanish (while training in an other language) the system seems better recognising newspapers rather than satire.

One of the most interesting result is the 5-fold cross validation over the whole dataset, including all the accounts of all the languages (last raw of Table 5). The F1 score of this experiment is 0.767 and it can be considered a high score considering the noise that could derive when we generate the same features in different languages. Indeed, the word-based model scores 8 point less.

Train	Test	Non-sat.	Satire	Mean
English	Spanish	0.676	0.475	0.575
English	Italian	0.71	0.555	0.632
Spanish	English	0.031	0.669	0.35
Spanish	Italian	0.657	0.733	0.695
Italian	Spanish	0.664	0.05	0.357
Italian	English	0.665	0.156	0.41
All word-based	*(5-folds)*	0.659	0.713	0.686
All intrinsic	*(5-folds)*	**0.765**	**0.769**	**0.767**

Table 5: Cross-Languages experiments. Train in one language and testing in a different one, and in the last two raws a 5-folds cross validation on the whole dataset (all accounts of all languages) using the word-based and the intrinsic-word models.

n	English	Spanish	Italian	Eng+Spa+Ita
1	**[char]**length	**[char]**length	**[char]**uppercase-ratio	**[char]**tot-char
2	**[pos]**num-noun	**[char]**uppercase-ratio	**[char]**lowercase-ratio	**[char]**uppercase-ratio
3	**[char]**first-uppercase	**[char]**lowercase-ratio	**[pos]**num-verbs	**[char]**lowercase-ratio
4	**[senti]**words-with-pol	**[char]**first-uppercase	**[char]**length	**[pos]**num-nouns
5	**[char]**lowercase-ratio	**[pos]**num-noun	**[amb]**verb-synset-mean	**[char]**first-uppercase
6	**[senti]**positive-ratio	**[char]**longest-word	**[amb]**verb-synset-gap	**[pos]**num-verbs
7	**[freq]**rarest-noun	**[char]**exclamation-mark	**[amb]**verb-max-synset	**[amb]**verb-max-synset
8	**[char]**uppercase-ratio	**[char]**longest-shortest-gap	**[freq]**verb-mean-freq	**[amb]**verb-synset-gap
9	**[senti]**noun-with-pol-ratio	**[char]**average-word-length	**[freq]**verb-gap-freq	**[amb]**verb-synset-mean
10	**[amb]**noun-synset-gap	**[pos]**num-adpositions	**[freq]**rarest-verb	**[freq]**rarest-noun
11	**[char]**shortest-word	**[freq]**noun-gap-freq	**[pos]**num-noun	**[char]**longest-shortest-gap
12	**[freq]**rarest-word	**[freq]**rarest-adjective	**[char]**longest-word	**[char]**longest-word
13	**[amb]**word-synset-gap	**[freq]**rarest-noun	**[char]**longest-shortest-gap	**[char]**average-word-length
14	**[amb]**max-noun-synset	**[pos]**num-numbers	**[amb]**max-num-synset	**[senti]**words-with-pol
15	**[syno]**lowest-gap	**[amb]**synset-mean	**[pos]**num-pronoun	**[freq]**verb-mean-freq

Table 6: Best 15 features of the language-independent model (all features without word-based) ranked considering the information gain scores in the N1+N2 vs S1+S2 dataset. In the last column are reported the best features considering the arithmetic average of the information gain of each language. In **[bold]** are reported the group of each feature.

6 Discussion

Across the three languages we considered, the different quality of the linguistic resources adopted as well as the distinct accuracy on the NLP tools exploited to analyse Tweets introduce some biases when we generate our set of cross lingual features, referred to as intrinsic-word features. These biases have to be considered in the interpretation of the results of our cross-lingual experiments.

Our intrinsic-word features model (features from 2 to 6 in Section 4) outperforms the word-based baseline in each single language experiments, showing that the use of intrinsic-word features represent a good approach for satire detection across the three languages we considered. The best performance of the intrinsic-word features occurs in the Italian dataset, where they obtain an F-measure of 0.746 in one combination, while the word-based model scores only 0.541. Adding word-based features to the intrinsic-word model seems to increase the performance only in the second type of experiment, where all accounts are included in the training. Yet, the word-based features are strictly related to the words used by specific accounts. The use of word-based features is not domain and language independent because it is strictly related to specific words rather than inner "cross-account" and "cross-language" linguistic traits of satire.

The best features (see Table 6) across the languages were Characters, Part Of Speech and Ambiguity. In English we note that beside the Characters features (relevant in all the languages), the number of words with polarity (positive or negative) is important (but not that important for Spanish and Italian). Additionally, the use of rare nouns (infrequent) is a characteristic of English satire. What distinguishes Spanish satire is the number of nouns and appositions, and the use of long words. In this language also the detection of rare nouns and rare adjective is a distinctive feature of satire. In Italian, the Characters feature are also important, especially the uppercase and lowercase ratio. Moreover in Italian satire verbs play a key role. Indeed the number of verbs, the number of synsets associated to a verb and the frequency usage of a verb (if it is rare or not) are strongly distinctive for Italian satirical news. Furthermore, as in Spanish, using long words may be sign of Italian satire. One last curious result is that the use of slang and bad words is not relevant if compared to the satire detection contributions of structural features (Characters and Frequency) and semantic features (like ambiguity). This fact suggests that the satirical news of the accounts we selected mimic appropriately non-satirical news.

In the cross-lingual experiments we can deduce that it is not always possible to train in one language and test in another one with the proposed model (Table 5). Yet, there are interesting results. For instance, when training in Italian the system is not able to detect English and Spanish satire, but when testing on Italian and training in the other languages results are better. The interpretation may be that Italian satire is less intricate, easy to detect but not able to recognise other kind of satire. Our intrinsic-word model when trained in Spanish is able to detect Italian satire with a precision of 0.695 (with satire F1 of 0.733), which is a very interesting result considering the complexity of the task. We need to consider that the two datasets are written in different languages, and the satirical topics are different (as they are related to politics and culture). On the other hand English can not be detected by Spanish nor Italian systems, but they both can recognise an aspect of the English dataset (Spanish recognises English satire, and Italian recognises with good accuracy, F1 of 0.71, English newspapers). Finally, the last results that deserve further analysis is the 5-fold cross validation over the all dataset, where all the accounts of all the languages were included. The accuracy of our model is promising (F1 of 0.767) as in this dataset the noise is very high: 22,228 Tweets on three different languages and different topics.

7 Conclusions

In this paper we proposed an approach to detect news satire in Twitter in different languages. Our approach avoids the use of word-based features (Bag Of Words), by relying only on language independent features, that we referred to as intrinsic-word features since they aim to detect inner characteristics of the words. We tested our approach on English, Spanish and Italian Tweets and obtained significant results. Our system was able to recognise if a Tweet advertises a non-satirical or satirical news, outperforming a word-based baseline. Moreover we tested the system with cross-language experiments, obtaining interesting results that deserve of a deeper investigation. We plan to explore our approach with new languages, and seek methods to combine languages to obtain better accuracy in cross-lingual satiric detection.

References

[Baccianella *et al.*, 2010] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 2200–2204, 2010.

[Barbieri and Saggion, 2014a] Francesco Barbieri and Horacio Saggion. Automatic Detection of Humour and Irony in Twitter. *International Conference on Computational Creativity, ICCC*, 2014.

[Barbieri and Saggion, 2014b] Francesco Barbieri and Horacio Saggion. Modelling Irony in Twitter. In *Proceedings of the EACL Student Research Workshop*, pages 56–64, Gothenburg, Sweden, April 2014. ACL.

[Barbieri *et al.*, 2014] Francesco Barbieri, Francesco Ronzano, and Horacio Saggion. Italian Irony Detection in Twitter: a First Approach. *The First Italian Conference on Computational Linguistics CLiC-it 2014*, page 28, 2014.

[Basile and Nissim, 2013] Valerio Basile and Malvina Nissim. Sentiment analysis on italian tweets. In *Proceedings of the 4th WASSA Workshop*, pages 100–107, 2013.

[Bontcheva *et al.*, 2013] Kalina Bontcheva, Leon Derczynski, Adam Funk, Mark A. Greenwood, Diana Maynard, and Niraj Aswani. TwitIE: An Open-Source Information Extraction Pipeline for Microblog Text. In *Proceedings of Recent Advances in Natural Language Processing Conferemce*, 2013.

[Burfoot and Baldwin, 2009] Clint Burfoot and Timothy Baldwin. Automatic satire detection: Are you having a laugh? In *Proceedings of the ACL-IJCNLP 2009 conference short papers*, pages 161–164. ACL, 2009.

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

[Carvalho *et al.*, 2009] Paula Carvalho, Luís Sarmento, Mário J Silva, and Eugénio de Oliveira. 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, 2009.

[Colletta, 2009] Lisa Colletta. Political satire and postmodern irony in the age of stephen colbert and jon stewart. *The Journal of Popular Culture*, 42(5):856–874, 2009.

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

[Knight, 2004] Charles A Knight. *The literature of satire*. Cambridge University Press, 2004.

[LaMarre *et al.*, 2009] Heather L LaMarre, Kristen D Landreville, and Michael A Beam. The irony of satire political ideology and the motivation to see what you want to see in the colbert report. *The International Journal of Press/Politics*, 14(2):212–231, 2009.

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

[Mann, 1973] Jill Mann. *Chaucer and Medieval Estates Satire: The Literature of Social Classes and the General Prologue to the Canterbury Tales*. Cambridge University Press Cambridge, 1973.

[Miller, 1995] George A Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[Pang and Lee, 2008] Bo Pang and Lillian Lee. Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, January 2008.

[Peter, 1956] John Peter. Complaint and satire in early english literature. 1956.

[Picard, 1997] RW Picard. *Affective Computing*. MIT Press, 1997.

[Platt, 1999] John Platt. Fast training of support vector machines using sequential minimal optimization. *Advances in kernel methodssupport vector learning*, 3, 1999.

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

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

[Tao and Tan, 2005] Jianhua Tao and Tieniu Tan. Affective computing: A review. In *Affective computing and intelligent interaction*, pages 981–995. Springer, 2005.

[Turney, 2002] Peter D Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th ACL*, pages 417–424. Association for Computational Linguistics, 2002.

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