Detecting Irony and Sarcasm in Microblogs: The Role of Expressive Signals and Ensemble Classifiers

Elisabetta Fersini
University of Milano-Bicocca
20126 - Milan, Italy
Email: fersini@disco.unimib.it

Federico Alberto Pozzi
University of Milano-Bicocca
20126 - Milan, Italy
Email: federico.pozzi@disco.unimib.it

Enza Messina
University of Milano-Bicocca
20126 - Milan, Italy
Email: messina@disco.unimib.it

***Abstract*—The automatic detection of sarcasm and irony in user generated contents is one of the most challenging task of Natural Language Processing. In this paper we address this problem by introducing Bayesian Model Averaging (BMA), an ensemble approach to take into account several classifiers according to their reliabilities and their marginal probability predictions. The impact of the most used expressive signals (pragmatic particles and POS tags) have been evaluated in baseline models (traditional classifiers and majority voting) as well as in the proposed BMA approach. Experimental results highlight two main findings: (1) not all the features are equally able to characterize sarcasm and irony and (2) BMA not only outperforms traditional state of the art models, but is also able to ensure notable generalization capabilities both on ironic and sarcastic text.**

I. Introduction

As defined in [1], a figure of speech is any artful deviation from the ordinary mode of speaking or writing. The most problematic figures of speech that Natural Language Processing (NLP) techniques try to detect are sarcasm and irony [2], which are commonly used to convey implicit criticism with a particular victim as its target, saying or writing the opposite of what the author means [3]. Although sarcasm and irony are a well-studied phenomenons in linguistics, psychology and cognitive science [4], [5], their automatic detection in NLP is still in its infancy because of its complexity. However, some preliminary investigations have been conducted. In [6], a semi-supervised learning approach was proposed to automatically expand the initial seed set of labeled messages through Web search. Terms and punctuation-based features have been used for training some baseline classifiers. A supervised approach has been proposed in [7], where the problem is studied in the context of Sentiment Analysis using Twitter data. The authors used unigrams, word categories, interjections (e. g., ah, yeah), and punctuation as features. Emoticons and ToUser (which marks if a tweet is a reply to another tweet) were also used. In [8] the authors perform an analysis of the effect of sarcasm scope on the polarity of messages, where sarcasm is explicitly available (i.e. through specific hashtags such as #irony and #sarcasm). In [9], a set of text features for recognizing irony at a linguistic level have been investigated. Finally, in [10] a supervised model has been exploited to distinguish sarcasm in English and Czech languages by using n-grams, patterns, POS tags, emoticons, punctuation and word case.

While the above mentioned investigation have shown promising results in terms of recognition performance (either on sarcasm or irony), less effort has been spent on the machine learning perspective and on the understanding of which linguistic feature contributes more on the characterization of sarcasm and irony. For this reason, in this paper we propose an ensemble approach - based on the Bayesian Model Averaging paradigm - which makes use of models trained using several linguistic features to better distinguish between sarcastic and non sarcastic statements, as well as ironic and non ironic text.

II. Ensemble approaches for sarcasm and irony

The state of the art about irony and sarcasm recognition basically comprises traditional supervised methods (such as Naive Bayes, Support Vector Machines and so on) to detect a specific figure of speech [10], [7], [9]. The common approach of these studies concerns the investigation of several linguistic features to train different classifiers, to subsequently identify the best model able to fit either sarcasm or irony statements. In particular, the state of the art studies, are based on the classical statistical inference paradigm, which selects the single model (over a set of hypothesis) with the highest likelihood given the training data and uses it to make predictions. This may lead to over-confident inferences and decisions that do not take into account the inherent uncertainty of the natural language in wider contexts such as social media. Instead, the idea behind an ensemble mechanism is to exploit the characteristics of several independent classifiers by combining them in order to achieve higher performance than the best single classifier. Two main ensemble approaches have been considered to detect sarcasm and irony, which are reported in the following subsections.

A. Majority Voting

One of the most popular ensemble system is *Majority Voting (MV)*, which is characterized by a set of "experts" that classifies the message figure of speech by considering the vote of each classifier as equally important and determines the final label (sarcasm/not sarcasm, irony/not irony) by selecting the most popular label prediction [11]. If the number of classifiers which label a tweet as *sarcastic* (or ironic) is strictly equal to the number of classifiers which label the same tweet as *not sarcastic* (or not ironic), the final label is determined according to the classifier that is able to ensure the highest accuracy. When combing multiple independent and diverse decisions each of which is at least more accurate than random guessing, random errors should be canceled out, reinforcing therefore correct decisions.

Copyright notice: 978-1-4673-8273-1/15/$31.00 2015 IEEE

B. Bayesian Model Averaging

The most important limit introduced by MV is that the models to be included in the ensemble have uniform distributed weights regardless their reliability. When dealing with sarcasm and irony, an ensemble should take into account the reliability of each model to detect rare statement (irony and sarcasm are much less frequent than others). To this purpose, the uncertainty left by data and models can be filtered by considering the Bayesian paradigm. In particular, through Bayesian Model Averaging (BMA) all possible models in the hypothesis space could be used when making predictions, considering their marginal prediction capabilities and their reliability. Given a dataset $\mathcal{D}$ and a set C of classifiers, BMA assigns to a message m the label $l(m)$ that maximizes:

$$P(l(m) \mid C, \mathcal{D}) = \sum_{i \in C} P(l(m) \mid i, \mathcal{D}) P(i \mid \mathcal{D}) \tag{1}$$

where $P(l(m) \mid i, \mathcal{D})$ is the marginal distribution of the label predicted by classifier i and $P(i \mid \mathcal{D})$ denotes the posterior probability of model i. The posterior $P(i \mid \mathcal{D})$ can be computed as:

$$P(i \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid i) P(i)}{\sum\limits_{j \in C} P(\mathcal{D} \mid j) P(j)} \tag{2}$$

where $P(i)$ is the prior probability of i and $P(\mathcal{D} \mid i)$ is the model likelihood. In Eq. (2), $\sum_{j \in C} P(\mathcal{D} \mid j) P(j)$ is assumed to be a constant and therefore can be omitted. Therefore, BMA assigns the label $l^{\text{BMA}}(m)$ to m as follows:

$$\begin{aligned} l^{\text{BMA}}(m) &= \arg\max_{l(m)} P(l(m)|C, \mathcal{D}) \\ &= \arg\max_{l(m)} \sum_{i \in C} P(l(m)|i, \mathcal{D}) P(i|\mathcal{D}) \\ &= \arg\max_{l(m)} \sum_{i \in C} P(l(m)|i, \mathcal{D}) P(\mathcal{D}|i) P(i) \end{aligned} \tag{3}$$

We proposed to replace the implicit measure $P(\mathcal{D} \mid i)$ by an explicit estimate, known as F_1-measure, obtained during a preliminary evaluation of the classifier i. In particular, by performing a cross validation, each classifier can produce an average measure stating how well a learning machine generalizes to unseen data. Note that $P(\mathcal{D}|i)$ computed on each fold separately, i. e. the estimation of $P(\mathcal{D}|i)$ used for inference in a particular fold has been performed on the remaining folds. This procedure has been adopted to avoid over-fitting. Considering ϕ-folds for cross validating a classifier i, the measure $P(\mathcal{D} \mid i)$ can be approximated as

$$P(\mathcal{D} \mid i) \approx \frac{1}{\iota} \sum_{\iota=1}^{\phi} \frac{2 \times P_{i\iota}(\mathcal{D}) \times R_{i\iota}(\mathcal{D})}{P_{i\iota}(\mathcal{D}) + R_{i\iota}(\mathcal{D})} \tag{4}$$

where $P_{i\iota}(\mathcal{D})$ and $R_{i\iota}(\mathcal{D})$ denote precision and recall obtained by classifier i in fold ι. In this way $P(l(m)|i, \mathcal{D})$ in Eq. (3) is tuned according to the ability of the classifier to fit the training data. This approach allows the uncertainty of each classifier to be taken into account, avoiding over-confident inferences.

A crucial issue of most ensemble methods is referred to the selection of the optimal set of models to be included in the final voting mechanism. In fact, including many powerful classifiers in an ensemble does not ensure better performance. To this purpose, an optimal ensemble S should minimize three main types of error that can be produced:

- *Bias*: error caused by an inaccurate model

  $$E[S] = \frac{1}{|D|} \sum_{m=1}^{|D|} f_e(l(m)^*, l(m)^{BMA})] \tag{5}$$

  where $f_e(l(m)^*, l(m)^{BMA})$ is an estimation of the error provided by BMA with respect to the correct label $l(m)^*$.

- *Variance*: error caused by the data sample

  $$\begin{aligned} Var(S) = \frac{1}{n^2} Var\Big(\sum_{i \in S} i\Big) = \\ = \frac{1}{n^2} \Big[\Big(\sum_{i \in S} Var(i)\Big) + 2 \sum_{i \in S} \sum_{\substack{j<i \\ j \in S}} Cov(i, j)\Big] \end{aligned} \tag{6}$$

  where n is the number of classifiers enclosed in the ensemble S, while $Var()$ and $Cov()$ denote variance and covariance of the models in the ensemble respectively.

- *Noise*: error on unseen samples

What counts for determining the optimal ensemble is the trade-off[1] given by:

$$\textit{Trade-Off} = E[S] + Var(S) \tag{7}$$

In this scenario, the variance of the ensemble decreases as the number of independent classifiers increases. On the contrary, if classifiers are dependent and in particularly positively correlated, the variance increases according to the number of models enclosed. For what concerns the bias, the higher is the prediction accuracy of the ensemble and the lower is the bias.

The composition of the optimal ensemble should therefore reduce variance to ensure results that are less dependent on peculiarities of a single training set, and reduce bias to guarantee a voting mechanism that is a more expressive than a single classifier. The selection of the optimal set of models to be enclosed in the final ensemble is as a result a combinatorial optimization problem over $\sum_{p=1}^{n} \frac{n!}{p!(n-p)!}$ possible solutions, where n is the number of classifiers and p represents the dimension of each potential ensemble. Several metrics have been proposed in the literature to evaluate the contribution of classifiers to be included in the ensemble (see [12]) in order to reduce the bias-variance trade-off. To the best of our knowledge these measures are not suitable for a Bayesian Ensemble, because they assume uniform weight distribution of classifiers. In this study, we propose a heuristic able to compute the discriminative marginal contribution that each classifier provides with respect to a given ensemble. In order to illustrate this strategy, consider a simple case with two classifiers named i and j.

[1] the noise term related to the "true relationship" between observation and label cannot fundamentally be reduced by any model, becoming therefore invariant.

To evaluate the contribution (gain) that the classifier i gives with respect to j, we need to introduce two cases:

1) j incorrectly labels the sentence s, but i correctly tags it. This is the most important contribution of i to the voting mechanism and represents how much i is able to correct j's predictions;
2) Both i and j correctly label s. In this case, i corroborates the hypothesis provided by j to correctly label the sentence.

On the other hand, i could also bias the ensemble prediction in the following cases:

3) j correctly labels sentence s, but i incorrectly tags it. This is the most harmful contribution in a voting mechanism and represents how much i is able to negatively change the (correct) label provided by j.
4) Both i and j incorrectly label s. In this case, i corroborates the hypothesis provided by j leading to a double misclassification of the sentence.

To formally represent the cases above, let compute $P(i = 1 \mid j = 0)$ as the number of instances correctly classified by i over the number of instances incorrectly classified by j (case 1) and $P(i = 1 \mid j = 1)$ the number of instances correctly classified both by i over the number of instances correctly classified by j (case 2). Analogously, let $P(i = 0 \mid j = 1)$ be the number of instances misclassified by i over the number of instances correctly classified by j (case 3) and $P(i = 0 \mid j = 0)$ the number of instances misclassified by i over the number of instances misclassified also by j (case 4).

The contribution r_i^S of each classifier i belonging to a given ensemble $S \subseteq C$ can be estimated as:

$$r_i^S = \frac{\sum\limits_{j \in \{S \setminus i\}} \sum\limits_{q \in \{0,1\}} P(i = 1 \mid j = q) P(j = q)}{\sum\limits_{j \in \{S \setminus i\}} \sum\limits_{q \in \{0,1\}} P(i = 0 \mid j = q) P(j = q)} \tag{8}$$

where $P(j = q)$ is the prior of classifier j to either correctly or incorrectly predict labels. In particular, $P(j = 1)$ denotes the percentage of correctly classified instances (i.e. accuracy), while $P(j = 0)$ represents the rate of misclassified (i.e. error rate).

Once the contribution of each classifier has been computed, a further issue to be addressed concerns with the search strategy for determining the optimal ensemble composition. The greedy approaches presented in the literature can be distinguished according to the search direction: *forward selection* ([13], [14]) and *backward elimination* ([15], [16]).

In *forward selection*, the initial ensemble S is an empty set. The algorithm iteratively adds to S the classifier $i \in \{C \setminus S\}$ that optimizes a given evaluation function. In *backward elimination*, the ensemble S initially contains all the classifiers of the complete set C and iteratively removes the classifier $i \in S$ that optimizes the evaluation function. The advantage of backward elimination is that recognizing irrelevant models is straightforward. Removing a relevant model from a complete set should cause a decline in the evaluation, while adding a relevant model to an incomplete set may have an immediate impact. According to this consideration, the proposed evaluation function r_i^S is included in a greedy strategy based on backward elimination: starting from an initial set $S = C$, the contribution r_i^S is iteratively computed excluding at each step the classifier that achieves the lowest r_i^S. The proposed strategy allows us to reduce the search space from $\sum_{p=1}^{n} \frac{n!}{p!(n-p)!}$ to $n - 1$ potential candidates for determining the optimal ensemble, because at each step the classifier with the lowest r_i^S is disregarded until the smallest combination is achieved.

Another issue that concerns greedy selection is the stop condition related to the search process, i.e. how many models should be included in the final ensemble. The most common approach is to perform the search until all models have been removed from the ensemble and select the sub-ensemble with the lowest error (or higher accuracy) on the evaluation set. Alternatively, other approaches select a fixed number of models (or a percentage of the original ensemble).

In this paper, we propose to perform a backward selection until a local maxima of average classifier contribution is achieved. In particular, the backward elimination will continue until the average classifier contribution (ACC) of a sub-ensemble with respect to the previous parent ensemble will decrease. Indeed, when the average contribution decreases the parent ensemble corresponds to a local maxima and therefore is accepted as optimal ensemble combination. More formally, an ensemble S is accepted as optimal composition if the following condition is satisfied:

$$\frac{ACC(S)}{|S|} \geq \frac{ACC(S \setminus x)}{|S - 1|} \tag{9}$$

where $ACC(S)$ is estimated as the average r_i^S over the classifiers belonging to the ensemble S. Note that the contribution of each classifier i is computed according to the ensemble S, that is iteratively updated once the worst classifier is removed. This leads to the definition of S characterized by a decreasing size ranging from $|S| = N, N - 1, \ldots, 1$.

In order to define the initial ensemble, the baseline classifiers in C are selected in order to exhibit some level of dissimilarity that could lead to a reduction of the bias-variance trade-off. This can be achieved using models that belong to different families (i.e. generative, discriminative and large-margin models). As general remarks, this diversity helps ensembles to better capture different patterns of the natural language. Once this requirement is satisfied, the baseline classifiers to be enclosed in an ensemble can be arbitrary selected.

III. Feature Expansion

The traditional feature vector representing a message m (used to train a given classifier) only includes terms that belong to a common vocabulary V of terms derived from a message collection:

$$\vec{m} = (w_1, w_2, ..., w_{|V|}, l) \tag{10}$$

where w_t denotes the weight of term t belonging to m with label l. However, some expressive signals, typical of microblogs and indicative of sarcastic and/or ironic statements, can be used to enhance the traditional feature vector and therefore learning models. The expanded feature vector of a message is defined as:

$$\vec{m}_{\text{new}} = (w_1, w_2, ..., w_{|V|}, f_1, f_2, \dots, f_n, l) \tag{11}$$

where $f_1, f_2, \dots, f_n$ represent the n additional features discussed in the following.

A. Pragmatic particles

To better capture non-literal meaning enclosed in textual messages, several valuable expressive forms should be taken into account. Pragmatic particles, such as emoticons, emphatic and onomatopoeic expressions, represent those linguistic elements typically used on social media to elicit a given message. Among them, the following pragmatic particles have been considered:

- *Emoticons* are introduced as non-verbal components into the written language, mirroring the role played by facial expressions in speech [17]. According to the assumption that there exists a relationship between the sentiment orientation of emoticons and sarcasm/irony, emoticons have been distinguished in two main categories, i. e. positive (e. g., ':-)', ':D') and negative (e. g., ':-(', '=(').
- *Initialisms for emphatic expressions* represent a further pragmatic element used in non-verbal communications in online social media. These emphatic abbreviations play a similar role of emoticons: expressions such as 'ROFL' (Rolling On Floor Laughing) clearly represent positive expressions, while abbreviations as 'BM' (Bad Manner) denote negative statements.
- *Onomatopoeic expressions* in online social media can help to convey sarcasm and irony, e. g., 'bleh' and 'wow'. These patterns have been detected by dictionaries a priori defined [18].
- *Punctuation* in online social media does not follow orthographic conventions, e. g., "*Oomf gone have such a lovely summer!!!!!*". For this reason, punctuation has been placed under pragmatic particles. Question and exclamation marks have been considered as possible indicators of sarcasm and irony.

B. Part-Of-Speech (POS) lexical components

We argue that also POS tags could be relevant indicators of sarcasm and irony. For this reason, a POS tagger has been applied in order to assign lexical functions (verb, noun, adjectives) to each term (the list of considered tags is reported in Table I). Since online conversational text differs markedly from traditional written genres like newswire, we used a supervised POS tagger[2] defined for Twitter messages [19]. The impact of POS tags and pragmatic particles has been validated on traditional baseline classifiers as well as on BMA traditional MV.

Nominal	**Other closed-class words**
N - common noun	D - determiner
O - pronoun (personal; not possessive)	P - pre- or postposition, or subordinating conjunction
ˆ- proper noun	& - coordinating conjunction
S - nominal + possessive	T - verb particle
Z - proper noun + possessive	X - existential there
Other open-class words	**Other compounds**
V - verb	L - nominal + verbal (e. g., i'm)
A - adjective	M - proper noun + verbal
R - adverb	Y - X + verbal
! - interjection	

TABLE I. LIST OF TAGS USED AS ADDITIONAL FEATURES

IV. EXPERIMENTS

A. Dataset and settings

In order to perform an experimental investigation, two dataset have been used. The first dataset, related to sarcastic statements, has been collected by downloading some posts from Twitter using specific hashtags. In particular, to automatically collect sarcastic tweets, we followed the approach introduced in [6], [7], i. e. we used hashtags that directly express sarcasm (#sarcasm, #sarcastic), following the idea that the best judge of whether a tweet is intended to be sarcastic is the author himself. However, this automatic annotation process does not guarantee that tweets without the #sarcasm or #sarcastic hashtags are effectively not sarcastic. To this purpose, we asked three researchers to perform a manual review of the tweets that have been automatically labeled. The inter-agreement between annotators has been computed according to the Fleiss' kappa statistics [20], which measures the reliability agreement of labeling over that which would be expected by chance (when multiple annotators are involved). In our case, the inter-agreement statistics $\kappa = 0.79$ indicates a substantial agreement among annotators.

The final dataset was composed of about 8000 tweets, for which two annotators over three agreed on the same label. In order to reduce over-fitting due the presence of posts on the same topic/event during a short period of crawling (2 days in our investigation), two datasets have been derived (4000 tweets each) by a random sampling. In the following we will refer to them as "sarcasm 1" and "sarcasm 2". The second dataset relates to a benchmark for irony detection purposes [9]. The dataset is composed of 10,000 tweets tagged as ironic and 30,000 as non-ironic (10,000 for each of the following topics: Education, Humour and Politics). With respect to the original paper, where this dataset has been presented and some preliminary experiments have been conducted, to perform the experiments we selected the most challenging task with unbalanced classes, i.e. to learn ironic vs others. Concerning the baseline classifiers to enclose in the ensembles, Multinomial Naive Bayes, Support Vector Machines, Bayesian Networks and Decision Trees have been considered. Probabilistic SVMs have been trained with linear kernel (with cost parameter equal to 1.0 and tolerance to misclassification equal to 0.0010). K2 search algorithm has been exploited to learn the structure of the Bayesian Network. For Decision Trees, C4.5 (J48 in Weka) has been adopted while for Multinomial Naive Bayes no particular setting is required.

In order to evaluate the performance achieved by the investigated approaches, a 10-folds cross validation has been

[2] www.ark.cs.cmu.edu/TweetNLP/

adopted. Regarding the weighting schema used for the bag of words, experiments on the datasets have returned higher results using frequency both for terms and the additional features[3]. For evaluation, we employed Accuracy, Precision, Recall and F-Measure. Unlike well-formed documents (e.g., reviews), the writing style and the lexicon of microblogging messages are widely varied and often highly ungrammatical. For this reason, the collected messages need a painstaking text normalization process. We removed URLs, mention tags (@), hashtags (#) and retweet (RT) symbols.

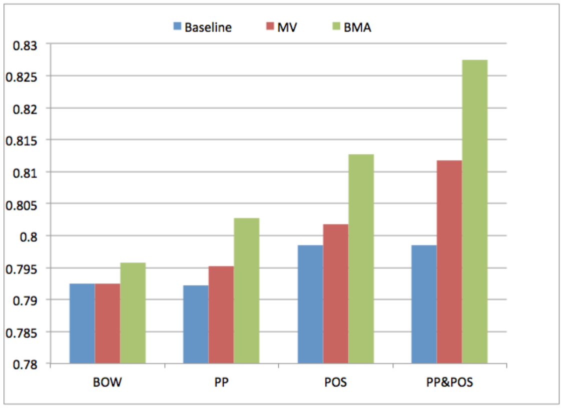

Fig. 1. Accuracy comparison among different configurations on sarcasm 1

B. Computational Results

In order to evaluate the proposed BMA approach we considered as (best) baseline classifier the one which the highest accuracy. We also considered four configurations: ***BOW*** (i. e. baseline setting where features are only terms), ***PP*** (i. e. terms + pragmatic particles), ***POS*** (i. e. terms + POS tags) and ***PP& POS*** (i. e. terms + pragmatic particles + POS tags).

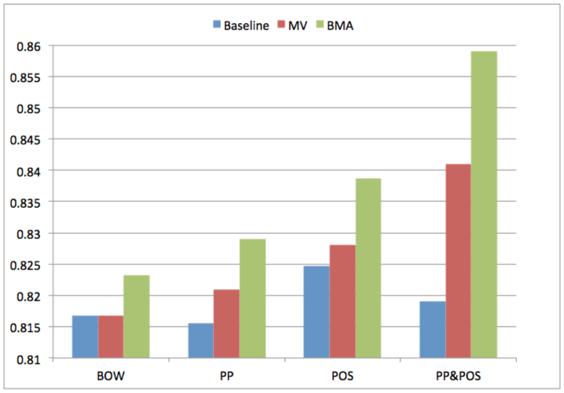

Fig. 2. Accuracy comparison among different configurations on sarcasm 2

By analyzing Figures 1 and 2 some considerations can be drawn. First of all, pragmatic particles do not provide any substantial improvement in the best baseline (i.e. Multinomial Naive Bayes) when considering sarcasm recognition tasks, while some increments in terms of accuracy are observed in the ensemble methods. This suggests that pragmatic particles contribute to improve the recognition ability of *poor classifiers* when sarcasm needs to be detected. It also emerges that pragmatic particles and POS tags independently lead to an increment of accuracy for MV and BMA, when pragmatic particles and POS tags are considered together, the performance is even higher. A further consideration relates BMA, which outperforms MV[4] and baseline classifiers in every configuration for all the studied datasets.

If we focus on Precision, Recall and F-Measure both for sarcastic and non sarcastic messages, we can grasp more peculiar behaviors of the considered features. We report in Tables II-VII the above mentioned measures distinguished in sarcastic (denoted by "+") and non sarcastic (represented by "-") messages, for the considered baseline methods as well as for the (best) ensembles (results are reported for the sarcasm 1 dataset, but analogous results have been obtained for sarcasm 2).

TABLE II. Impact of features on BMA - sarcasm

BMA				
	BOW	**PP**	**POS**	**PP & POS**
P+	**0.7709**	0.7718	0.7605	0.7697
R+	0.8245	0.8375	0.8540	**0.8548**
F+	0.7968	0.8033	0.8045	**0.8100**
P-	0.8114	0.8229	0.8335	**0.8339**
R-	0.7550	**0.7530**	0.7310	**0.7530**
F-	0.7822	0.7864	0.7789	**0.7914**

In bolds are reported the highest performance, while reduction of performance (of a given feature) with respect the the BOW representation are marked as underlined. Concerning BMA (Table II) we can notice that the proposed model is able to guarantee a good performance improvement, both for sarcastic and non sarcastic posts, when all the additional features are considered in the models that belong to the optimal ensemble (i.e. SVM+MNB). More precisely, for Precision of sarcastic posts, the best performance are achieved when the simple BOW representation is adopted, while for all the other measures the best performance are obtained when both PP and POS are exploited. This behavior suggests that the additional features have more impact on precision and recall of non sarcastic tweet and on recall of the sarcastic ones. When all additional features are considered, although a decreasing performance on $P+$ is obtained, the global measures of $F+$ and F- still improve.

TABLE III. Impact of features on MV - sarcasm

MV				
	BOW	**PP**	**POS**	**PP & POS**
P+	0.7616	**0.7712**	0.7633	0.7668
R+	0.8330	0.8360	**0.8610**	0.8550
F+	0.7957	0.8023	**0.8092**	0.8085
P-	0.8163	0.8210	**0.8406**	0.8362
R-	0.7400	**0.7520**	0.7330	0.7400
F-	0.7763	**0.7851**	0.7831	**0.7850**

Different results can be observed for Majority Voting (Table III). In this case, although pragmatic particles and POS contribute to improve the recognition performance of both sarcastic and non sarcastic posts, when features are jointly considered the optimal performance are not achieved.

[3] Frequency has been compared with the Boolean representation.

[4] T-Test rejects $H_0 : \mu_{\text{BMA}} - \mu_{\text{MV}} = 0$, where the critical region is $T > 2.92$ and $T = 3.08$ with $\alpha = 0.05$. Then the test does not reject $H_1 : \mu_{\text{BMA}} - \mu_{\text{MV}} > 0$.

TABLE IV. IMPACT OF FEATURES ON DT - SARCASM

	DT			
	BOW	**PP**	**POS**	**PP & POS**
P+	0.7104	0.7241	0.7429	**0.7431**
R+	0.7161	0.7310	0.7295	**0.7450**
F+	0.7133	0.7275	0.7361	**0.7441**
P-	0.7147	0.7284	0.7343	**0.7444**
R-	0.7090	0.7215	**0.7475**	0.7425
F-	0.7118	0.7249	0.7408	**0.7434**

Analogously to BMA, POS contributes more on sarcastic messages, while PP enhance the recognition of non sarcastic posts. This behavior is likely originated by the voting mechanism adopted by MV. In fact, as outlined in section II-B, including good classifiers in an ensemble does not ensure better performance (all the models enclosed in the ensemble should be i.i.d. and show different recognition abilities to reduce the bias-variance trade-off).

TABLE V. IMPACT OF FEATURES ON MNB - SARCASM

	MNB			
	BOW	**PP**	**POS**	**PP & POS**
P+	**0.7671**	0.7641	0.7635	0.7656
R+	0.8390	0.8455	**0.8650**	0.8605
F+	0.8014	0.8028	**0.8111**	0.8103
P-	0.8229	0.8271	**0.8443**	0.8408
R-	**0.7460**	0.7390	0.7320	0.7365
F-	0.7826	0.7806	0.7841	**0.7852**

If we focus on the baseline classifiers (Tables IV-VII), some considerations can be drawn. While Decision Trees and Support Vector Machines are positively affected by both PP and POS (and even more when jointly considered), Multinomial Naive Bayes and Bayesian Networks show a more complex behavior with respect to the considered features.

TABLE VI. IMPACT OF FEATURES ON SVM - SARCASM

	SVM			
	BOW	**PP**	**POS**	**PP & POS**
P+	0.7672	0.7710	0.7765	**0.7800**
R+	0.7568	0.7660	0.7990	**0.8015**
F+	0.7619	0.7685	0.7876	**0.7906**
P-	0.7607	0.7675	0.7930	**0.7959**
R-	0.7710	0.7725	0.7710	**0.7740**
F-	0.7658	0.7700	0.7813	**0.7848**

While in Multinomial Naive Bayes POS contributes more to recognize sarcastic messages, in Bayesian Networks the combination of PP and POS ensures an improvement both for sarcastic and non sarcastic posts.

TABLE VII. IMPACT OF FEATURES ON BN - SARCASM

	BN			
	BOW	**PP**	**POS**	**PP & POS**
P+	0.6692	**0.6965**	0.6824	0.6877
R+	0.7588	0.7150	0.7970	**0.8015**
F+	0.7112	0.7057	0.7352	**0.7402**
P-	0.7224	0.7072	0.7560	**0.7621**
R-	0.6260	**0.6885**	0.6290	0.6450
F-	0.6708	0.6977	0.6867	**0.6987**

As general remark on sarcasm, we can conclude stating that although most of the features positively contribute in terms of accuracy both in traditional baseline classifiers and in ensemble methods, some features are able to characterize more sarcastic messages. In particular, PP&POS contribute more on Decision Trees, Support Vector Machines and ensemble methods for both sarcastic and non sarcastic posts, while POS is able to characterize more sarcastic messages in MNB and BN.

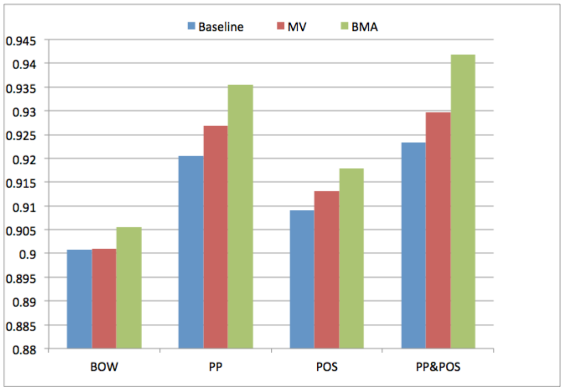

Fig. 3. Accuracy comparison among different configurations on irony

If we focus on irony statements (see Figure 3), we can point out that - on the opposite with respect to sarcastic posts - ironic messages are mainly captured by pragmatic particles (although POS tagging contributes with respect to the simple BOW model). The proposed BMA, where models are enriched by both pragmatic particles and POS tags, leads to notable accuracy performance.

In order to compare the proposed BMA with the state of the art on irony, we compared in Table VIII our approach to the one presented in [9]. Although the dataset used is strongly unbalanced, the proposed solution is able to deal with language uncertainty and real data distribution (where ironic statements represent a minority class), leading to remarkable results in terms of Accuracy, Precision, Recall and F-Measure.

	Accuracy	**Precision**	**Recall**	**F-Measure**
Reyes et al. [9]	0.8044	0.6610	0.4470	0.5330
BMA	**0.9417**	**0.7814**	**0.8987**	**0.8359**

TABLE VIII. PERFORMANCE COMPARISON ON IRONY (SEE THE ORIGINAL PAPER [9] AT PAGE 255)

Since the dataset is strongly unbalanced, we analyzed the performance also in terms of Precision, Recall and F-Measure distinguishing among ironic and non ironic posts. In Tables IX-XIV, we report the results for the ensembles as well for the baseline classifiers.

TABLE IX. IMPACT OF FEATURES ON BMA - IRONY

	BMA			
	BOW	**PP**	**POS**	**PP & POS**
P+	**0.9218**	0.8988	0.8409	0.8986
R+	0.6900	0.7766	0.7282	**0.7816**
F+	0.7927	0.8333	0.7805	**0.8360**
P-	0.9061	0.9288	0.9133	**0.9302**
R-	0.9700	0.9709	0.9541	**0.9800**
F-	0.9418	0.9493	0.9332	**0.9500**

Concerning BMA, although the dataset is strongly unbalanced, we can notice that the proposed model is able to guarantee remarkable performance improvement not only on non ironic messages, but also on the ironic ones. The best performance are obtained when all the additional features are considered in the models that belongs to the optimal ensemble (i.e. DT+SVM+BN). Differently from sarcasm, POS tags (considered as single additional feature) negatively contributes to the performance with respect to most of the measures.

TABLE X. IMPACT OF FEATURES ON MV - IRONY

MV				
	BOW	PP	POS	PP & POS
P+	0.7721	**0.8190**	0.5727	0.7812
R+	0.7286	**0.7842**	0.7392	0.7787
F+	0.7498	**0.8012**	0.6454	0.7799
P-	0.9112	**0.9291**	0.9161	0.9263
R-	0.9283	**0.9422**	0.8377	0.9273
F-	0.9197	**0.9356**	0.8752	0.9268

In this case, the most relevant features that contribute more to the optimal results are Pragmatic Particles. The positive contribution of PP can be also observed in MV, but more importantly in DT, SVM and BN.

TABLE XI. IMPACT OF FEATURES ON DT - IRONY

DT				
	BOW	PP	POS	PP & POS
P+	0.7915	0.8745	0.5710	**0.8714**
R+	0.6544	0.7967	0.6612	**0.7989**
F+	0.7165	**0.8338**	0.6128	0.8335
P-	0.8911	**0.9348**	0.8944	0.9342
R-	0.9425	**0.9619**	0.8525	0.9607
F-	0.9161	**0.9478**	0.8729	0.9476

TABLE XII. IMPACT OF FEATURES ON NB - IRONY

NB				
	BOW	PP	POS	PP & POS
P+	**0.6199**	0.6161	0.4837	0.5882
R+	**0.7157**	0.6975	0.6699	0.6718
F+	**0.6644**	0.6543	0.5618	0.6273
P-	**0.9001**	0.8945	0.8866	0.8852
R-	**0.8538**	0.8552	0.7831	0.8433
F-	**0.8763**	0.8744	0.8317	0.8637

TABLE XIII. IMPACT OF FEATURES ON SVM - IRONY

SVM				
	BOW	PP	POS	PP & POS
P+	0.8712	0.8915	0.6423	**0.8959**
R+	0.7076	0.7756	0.7349	**0.7804**
F+	0.7809	0.8295	0.6855	**0.8341**
P-	0.9083	0.9283	0.9197	**0.9298**
R-	0.9651	0.9685	0.8812	**0.9698**
F-	0.9359	0.9480	0.9000	**0.9494**

A completely different behavior can be observed for the Naive Bayes model. In this case, all the considered features negatively affect the prediction of both ironic and non ironic messages. In fact, in this case, the best performance are achieved when the simple BOW representation is adopted.

As general remark on irony, we can conclude stating that although most of the features positively contribute in terms of accuracy both in traditional baseline classifiers and in ensemble methods, Pragmatic Particles represent the most discriminative features that contribute more to the achievement of the optimal results (usually obtained when enriching BOW with both PP and POS).

TABLE XIV. IMPACT OF FEATURES ON BN - IRONY

BN				
	BOW	PP	POS	PP & POS
P+	0.7680	**0.7992**	0.5727	0.7635
R+	0.7522	**0.8273**	0.7630	0.8017
F+	0.7600	**0.8130**	0.6543	0.7821
P-	0.9180	**0.9417**	0.9231	0.9328
R-	0.9243	**0.9307**	0.8333	0.9172
F-	0.9211	**0.9362**	0.8759	0.9249

Concerning the BMA model, a further analysis has been conducted to evaluate if the proposed selection strategy is able to reduce the bias-variance trade-off when dealing with figurative language. If we focus on sarcasm statements (sarcasm 1 dataset), the optimal ensembles for BOW, POS and PP&POS are the one composed of SVM+MNB, while for PP the optimal set is SVM+MNB+BN. If we consider the irony statements, the optimal ensembles determined for each additional feature and according to the proposed selection strategy are:

- BOW: DT+SVM+BN
- PP: DT+BN
- POS: SVM+BN
- PP&POS: DT+SVM+BN

In Tables XV and XVI, the generalization error (in terms of bias-variance trade-off) is reported for the considered figures of speech for all the considered features and for all the possible ensemble composition. We can easily note that the best generalization capabilities (i.e. the ones characterized by the lowest bias-variance trade-off) are achieved by the ensemble compositions derived by the proposed selection strategy[5].

TABLE XV. VARIANCE-BIAS TRADE-OFF ON SARCASM 1

BMA	BOW	PP	POS	PP&POS
DT, SVM, MNB, BN	0.3708	0.3610	0.3688	0.3891
DT, MNB, BN	0.3865	0.3738	0.3712	0.3900
DT, SVM, BN	0.4030	0.4003	0.3812	0.4012
SVM, MNB, BN	0.3729	**0.3537**	0.3837	0.3865
DT, SVM, MNB	0.3885	0.3799	0.4045	0.4100
DT, SVM	0.4311	0.4310	0.4299	0.4286
DT, MNB	0.4084	0.3967	0.4054	0.4123
DT, BN	0.4600	0.4467	0.4363	0.4290
SVM, MNB	**0.3581**	0.4135	**0.3642**	**0.3556**
SVM, BN	0.3995	0.4037	0.4211	0.4183
MNB, BN	0.3881	0.3940	0.4171	0.4166

TABLE XVI. VARIANCE-BIAS TRADE-OFF ON IRONY

BMA	BOW	PP	POS	PP&POS
DT, SVM, MNB, BN	0.2370	0.2299	0.2434	0.2172
DT, MNB, BN	0.2573	0.2335	0.2707	0.2432
DT, SVM, BN	**0.2330**	0.2276	0.2482	**0.2090**
SVM, MNB, BN	0.2531	0.2376	0.2646	0.2509
DT, SVM, MNB	0.2375	0.2287	0.2431	0.2141
DT, SVM	0.2426	0.2317	0.2468	0.2300
DT, MNB	0.2687	0.2361	0.2789	0.2348
DT, BN	0.2595	**0.2259**	0.2672	0.2329
SVM, MNB	0.2514	0.2366	0.2487	0.2347
SVM, BN	0.2540	0.2430	**0.2428**	0.2398
MNB, BN	0.2999	0.2800	0.3250	0.3021

As general conclusion, we can affirm that the proposed BMA model, together with the selection strategy, is able to define an ensemble composition that not only outperforms baseline classifiers and majority voting, but is also optimal in terms of generalization capabilities.

[5] Analogous results have been obtained also for the datasets of sarcasm 2

V. Conclusion and Future Work

In this work, we proposed an ensemble method based on Bayesian Model Averaging aimed at sarcasm and irony detection. We enhanced the traditional bag of word model by including several indications about the expressive signals typically used in microblogs. The experimental results show that the proposed solution outperforms the traditional classifiers and the well known Majority Voting mechanism. Regarding the considered expressive signals, not all the features are equally able to characterize sarcasm and irony. In particular, while sarcasm can be better characterized by POS tags, ironic statements are captured by Pragmatic Particles. Concerning the future works, the natural prosecution of the present study relates to Sentiment Analysis. Sarcasm and irony detection have been NLP tasks, but in recent years they have acquired particular importance when polarity classification tasks need to be addressed [21]. For this reason, we are considering a hierarchical voting framework where the discrimination between non-literal meanings is firstly addressed, to then approach the sentiment classification task.

References

[1] E. P. J. Corbett, *Classical Rhetoric for the Modern Student*, 2nd ed. Oxford University Press, 1971.

[2] A. N. Katz, H. Colston, and A. Katz, "Discourse and sociocultural factors in understanding nonliteral language," *Figurative language comprehension: Social and cultural influences*, pp. 183–207, 2005.

[3] S. McDonald, "Exploring the process of inference generation in sarcasm: A review of normal and clinical studies," *Brain and Language*, vol. 68, no. 3, pp. 486–506, 1999.

[4] R. W. Gibbs and H. L. Colston, *Irony in Language and Thought: A Cognitive Science Reader*. Lawrence Erlbaum Associates, 2007.

[5] A. Utsumi, "Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony," *Journal of Pragmatics*, vol. 32, no. 12, pp. 1777–1806, 2000.

[6] D. Davidov, O. Tsur, and A. Rappoport, "Semi-supervised recognition of sarcastic sentences in twitter and amazon," in *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, ser. CoNLL '10. Association for Computational Linguistics, 2010, pp. 107–116.

[7] R. González-Ibáñez, S. Muresan, and N. Wacholder, "Identifying sarcasm in twitter: A closer look," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, ser. HLT '11. Association for Computational Linguistics, 2011, pp. 581–586.

[8] D. Maynard and M. A. Greenwood, "Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis," in *Proceedings of LREC*, 2014.

[9] A. Reyes, P. Rosso, and T. Veale, "A multidimensional approach for detecting irony in twitter," *Language Resources and Evaluation*, vol. 47, no. 1, pp. 239–268, 2013.

[10] T. Ptácek, I. Habernal, and J. Hong, "Sarcasm detection on czech and english twitter," in *25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 213–223.

[11] T. G. Dietterich, "Ensemble learning," in *The Handbook of Brain Theory and Neural Networks*. Mit Pr, 2002, pp. 405–508.

[12] I. Partalas, G. Tsoumakas, and I. Vlahavas, "An ensemble uncertainty aware measure for directed hill climbing ensemble pruning," *Machine Learning*, vol. 81, no. 3, pp. 257–282, 2010.

[13] W. Fan, F. Chu, H. Wang, and P. S. Yu, "Pruning and dynamic scheduling of cost-sensitive ensembles," in *Eighteenth National Conf. on Artificial Intelligence*, 2002, pp. 146–151.

[14] G. Martínez-Muñoz and A. Suárez, "Pruning in ordered bagging ensembles," in *Proceedings of the 23^{rd} International Conference on Machine Learning*, 2006, pp. 609–616.

[15] R. E. Banfield, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer, "Ensemble diversity measures and their application to thinning," *Information Fusion*, vol. 6, p. 2005, 2004.

[16] R. Caruana, A. Niculescu-Mizil, G. Crew, and A. Ksikes, "Ensemble selection from libraries of models," in *Proceedings of the 21^{st} International Conference on Machine Learning*, 2004, pp. 18–25.

[17] J. B. Walther and K. P. D'addario, "The impacts of emoticons on message interpretation in computer-mediated communication," *Social Science Computer Review*, vol. 19, pp. 324–347, 2001.

[18] F. A. Pozzi, E. Fersini, E. Messina, and D. Blanc, "Enhance polarity classification on social media through sentiment-based feature expansion," in *WOA@AI*IA*, 2013, pp. 78–84.

[19] O. Owoputi, B. OConnor, C. Dyer, K. Gimpel, N. Schneider, and N. A. Smith, "Improved part-of-speech tagging for online conversational text with word clusters," *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 380–390, 2013.

[20] J. L. Fleiss, "Measuring nominal scale agreement among many raters." *Psychological bulletin*, vol. 76, no. 5, p. 378, 1971.

[21] A. Ghosh, G. Li, T. Veale, P. Rosso, E. Shutova, A. Reyes, and J. Barnden, "Semeval-2015 task 11: Sentiment analysis of figurative language in twitter," in *Int. Workshop on Semantic Evaluation (SemEval-2015)*, 2015.