# INAOE-UPV at SemEval-2018 Task 3: An Ensemble Approach for Irony Detection in Twitter

**Delia Irazú Hernández Farías**[1], **Fernando Sánchez-Vega**[1]
**Manuel Montes-y-Gómez**[1,2], and **Paolo Rosso**[2]
[1] Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Mexico
[2] PRHLT Research Center, Universitat Politècnica de València, Spain
{dirazuherfa,fer.callotl,mmontesg}@inaoep.mx
prosso@dsic.upv.es

## Abstract

This paper describes an ensemble approach to the SemEval-2018 Task 3. The proposed method is composed of two renowned methods in text classification together with a novel approach for capturing ironic content by exploiting a tailored lexicon for irony detection. We experimented with different ensemble settings. The obtained results show that our method has a good performance for detecting the presence of ironic content in Twitter.

## 1 Introduction

Social media provide a perfect scenario for exploiting language beyond its literal sense by using figurative language devices such as, for example, irony. Correctly identifying the real intention behind user-generated content is a big challenge for different areas related to computational linguistics. For example, in Sentiment Analysis (SA), the presence of irony could undermine the performance of systems dedicated to this task (Hernández Farías and Rosso, 2016). There are several disciplines studying irony from different perspectives. The most prevalent definition is that from Grice (1975), stating that the function of irony is to effectively communicate the opposite of the literal interpretation a given utterance.

Nowadays, with the growing interest in irony detection, there are several approaches[1] for addressing such an interesting task. Probably, the most widely used is that exploiting characteristics extracted from the text (such as n-grams, punctuation marks, part-of-speech labels, among others) on its own (Riloff et al., 2013; Ptáček et al., 2014). Inherent aspects of irony such as its very subjective component have also been considered (Reyes et al., 2013; Barbieri et al., 2014; Hernández Farías et al., 2016). Other methods have opted for taking advantage of information coming from the context in which a given utterance is produced (Rajadesingan et al., 2015). There are also some approaches exploiting deep learning techniques and word embeddings (Poria et al., 2016; Ghosh and Veale, 2016; Joshi et al., 2016; Nozza et al., 2016). A less explored strategy for addressing irony detection is the use of ensemble methods. Fersini et al. (2015) and Liu et al. (2014) compared the performance of ensemble approaches against traditional classifiers; the best results were obtained by the ensemble strategy setting.

In this paper we describe our participation to the SemEval-2018 Task 3: Irony detection in English tweets (Van Hee et al., 2018). The INAOE-UPV system explores the use of an ensemble approach that considers different combinations of three methods. The main contribution of our approach lies on the use of a list of potentially ironic and non-ironic terms in order to identify irony in tweets.

## 2 Method Description

In order to determine the presence of ironic content in tweets, we propose an ensemble of different methods, namely, a bag-of-words and word embeddings classifiers, as well as a voting scheme based on a list of potentially ironic and non-ironic terms.

### 2.1 Individual classifiers

#### Ironic/nonironic Orientation (irO)

This approach attempts to capture the *ironic* and *non-ironic* connotation of the words in a tweet in order to identify the presence of ironic content. Building a lexicon for irony detection is not a trivial task. It has been recognized in (Nozza et al., 2016) that a lexicon for irony detection can be derived by using a huge amount of data.

[1] For a more comprehensive overview of irony detection, see (Joshi et al., 2017).



*Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*, pages 594–599
New Orleans, Louisiana, June 5–6, 2018. 

To develop a lexicon for irony detection it is needed to calculate how much a word could be associated with an ironic or non-ironic sense. A widely exploited measure in SA for developing lexica is the Pointwise Mutual Information (PMI) (Church and Hanks, 1990). We decided to adopt a similar strategy to generate two lists of terms associated to ironic and non-ironic senses. As starting point we took advantage of a set of corpora from the state of the art in irony detection (henceforth *benchmark-corpora*). The datasets we used are described in (Reyes et al., 2013; Riloff et al., 2013; Barbieri et al., 2014; Ptáček et al., 2014; Mohammad et al., 2015; Ghosh et al., 2015; Sulis et al., 2016; Karoui et al., 2017). Overall, more than 165,000 tweets were used to generate the lists of words: $ironic\_terms$ and $nonironic\_terms$. We calculate the PMI score for each term[2] in the *benchmark-corpora*. After that, we selected only those terms with a PMI score greater than zero.

In order to determine the class of an instance we assigned a vote ($v$) for each word ($w$) in a given tweet ($t$). First, we filter out the stopwords in each tweet. Then, we search for the most similar term in each of the lists in order to determine whether $w$ is more related to an ironic or non-ironic sense. Mainly we compute a score that indicates the higher cosine similarity[3] among $w$ and each of the $N$ terms defined in our lists of words. As expected, the score for the words in $t$ that are directly included in $ironic\_terms$ or $nonironic\_terms$ will be 1.

$$simIro(w) = \max_{j=1...N}(sim(w, ironic\_terms_j))$$

$$simNI(w) = \max_{j=1...N}(sim(w, nonironic\_terms_j))$$

After this, the vote $v(w)$ is assigned according to the following criterion:

$$v(w) = \begin{cases} simIro & \text{if } simIro > simNI \\ -simNI & \text{if } simIro < simNI \end{cases} \quad (1)$$

Finally, the class of a tweet is determined by the sum of the votes from all words in $t$.

$$class(t) = \begin{cases} \text{irony} & \text{if } \sum_{i=1}^{|t|} v(w \in t) \geq 0 \\ \text{non-irony} & otherwise \end{cases}$$

[2]We removed those terms that occurred less than five times in each class.

[3]We calculated the cosine similarity exploiting pre-trained word vectors from the Google News Corpus.

**Bag-of-words based classifier (BOW)**

This approach is based on a bag-of-words (Salton et al., 1975) representation of the tweets. It uses unigrams as binary features. For the classification it employs a SVM classifier[4]. From here, we will use the acronym **BOW** to refer the use of the aforementioned individual approach.

**Word Embeddings based classifier (wEmb)**

This approach is based on the use of word embeddings. Particularly, it employs embeddings pre-trained on the Google News corpus (Mikolov et al., 2013) using the Continuous Bag-of-Words (CBOW) model[5]. In this case, tweets are represented by the centroid of the vectors from their words. Similar to the BOW approach, the classification is done by a SVM classifier. From now on the acronym **wEmb** will be used to refer to this approach.

## 2.2 Ensemble approaches for irony detection

We explored the use of different techniques relying on the words content in each tweet in order to identify the presence of irony. Each of the techniques we exploited has its own advantages and limitations. The **BOW** model allows to capture the existing topics in the vocabulary as well as discursive markers used in an ironic writing style. On the other hand, **wEmb** makes possible to catch abstract semantics of the words regardless of the available data for the task. With respect to **irO**, it attempts to simulate the interpretative process carried out to understand the ironic intention. Irony comprehension at an initial stage involves getting the literal sense of words (Giora and Fein, 1999) and then recognizing the figurative intention behind them. Thus, our method quantifies how many words are likely to be used in a literal or figurative sense before deciding whether a tweet is ironic or not. By proposing an ensemble using all the methods together we attempt to encompass different aspects of the use of vocabulary when the ironic phenomenon is present. Below, we introduce some

[4]We employed the SVM implementation of Weka (Hall et al., 2009).

[5]We also experimented with word embeddings trained using the benchmark corpora obtaining lower results than with Google News embeddings. Therefore, in order to participate in the shared task we decided to include only the latest kinds of embeddings.

ensemble approaches[6] proposed for capturing the presence of irony in Twitter.

**Coverage-based ensemble (ENS_cov)**

It is composed by BOW and wEmb. In Twitter data, there are many terms such as mentions, hashtags, emoji, URL, etc., that are unlikely to have an embedding. However, such kinds of terms are indeed covered by a model like BOW. To take advantage of both methods, we decided to combine them by considering a simple criterion depending on the coverage rate of the word embeddings ($cov\_emb$) in each single tweet. That is, if the $cov\_emb$ is greater than 75%, the tweet will be classified by the wEmb model, otherwise the decision will be made by the BOW approach.

**Majority vote ensemble (ENS_vot)**

In this approach, the decisions from the three individual methods (irO, BOW and wEmb) are combined following a majority vote strategy.

## 3 Experiments and Results

### 3.1 Task Description

This year, as part of SemEval-2018 the Task 3 on *Irony detection in English tweets* (Van Hee et al., 2018), was dedicated to the identification of ironic content in Twitter. The task is composed by two subtasks: ***Task A. Ironic vs. non-ironic***, the aim was to identify whether a tweet contains an ironic intention or not. The objective of the second one, ***Task B. Different types of irony***, was to classify a tweet in one out of four classes: (i) verbal irony realized through a polarity contrast, (ii) other verbal irony, (iii) situational irony, and (iv) non ironic. Participants were allowed to submit two different kinds of systems: *Constrained (C)* where only data provided for the task were used for training purposes, and *Unconstrained (U)* where additional data were exploited.

#### 3.1.1 Task A

In order to address Task A, we applied two different ensemble approaches. Our first submission was based on the coverage-based ensemble using a constrained setting (henceforth *taskA_ENS_cov_C*).

The second submission (henceforth *taskA_ENS_vot_U*) used the majority vote ensemble built on an unconstrained setting. BOW and wEmb models were trained by using only the training set provided by the organizers. Instead, irO involves the use of the benchmark corpora. Additionally, we collected a set of tweets containing the hashtags #irony and #sarcasm during the 2016 US Elections week[7] as well as the training data provided for the task for building the lists of ironic and non-ironic terms.

For experimental purposes, we applied a three fold cross-validation using the training data during the developing phase of the shared task. Table 1 shows the obtained results in $F_1$-Score.

| Method | $F_1$-Score |
|---|---|
| BOW | 0.62 |
| wEmb | 0.64 |
| irO | 0.63 |
| *taskA_ENS_cov_C* | 0.63 |
| *taskA_ENS_vot_U* | 0.65 |

Table 1: Results during the developing phase in Task A.

First, we evaluated each of the methods described in Section 2 individually (the first three rows in Table 1). The first two rows present the obtained results when the performance of BOW and wEmb was assessed using only the training data. Meanwhile, irO exploits both data from the task and external data. The highest result was achieved by the wEmb model. Despite being a basic method for identifying irony in tweets, our proposed approach (irO) achieves good performance even in comparison to powerful techniques such as word embeddings. Regarding the ensemble approaches, the best performance was reached by the majority vote approach.

#### 3.1.2 Task B

In order to address the Task B, we employed two different configurations of the majority vote approach (henceforth *taskB_ENS_vot_U1* and *taskB_ENS_vot_U2*), adding an additional criterion: in both cases, when the result of irO[8] indicates the presence of irony, we assigned one of the ironic-related classes by exploiting three different lists of words (one for each class in Task B) created following the same strategy described in Section 2.1. For *taskB_ENS_vot_U1*, the BOW and wEmb models were trained using the four classes in Task B; while in *taskB_ENS_vot_U2* four binary classifiers considering the combinations between

[6] Due to the lack of space we are not reporting all the experiments carried out.

[7] From 8th up to 18th November 2016.

[8] In this setting we also considered the corpora of the state of the art.

ironic classes and the non-ironic class in Task B. A weighted voting strategy was adopted in both ensembles. Table 2 shows the obtained results.

| Method | $F_1$-Score |
|---|---|
| BOW | 0.48 |
| wEmb | 0.31 |
| irO | 0.41 |
| *taskB_ENS_vot_U1* | 0.44 |
| *taskB_ENS_vot_U2* | 0.46 |

Table 2: Results during the developing phase in Task B.

The three methods were also evaluated individually for Task B. As it can be noticed, the best performance was achieved by BOW. The irO method performs better than wEmb. This is probably due to the fact of having few data for training the classifier. Neither of the ensemble methods improves the baseline, i.e., the BOW results.

### 3.2 Official Results

Table 3 shows the obtained results according to the official ranking of the shared task.

| Task | Method | F1-score |
|---|---|---|
| **A** | *taskA_ENS_cov_C* | **0.6265** |
| | *taskA_ENS_vot_U* | 0.6184 |
| **B** | *taskB_ENS_vot_U1* | 0.3497 |
| | *taskB_ENS_vot_U2* | 0.2148 |

Table 3: Official results obtained by our runs at the shared task. The underlined values are those in the official ranking of the task.

Our best result was in the constrained version of Task A (we ranked in the $11^{th}$ position). Regarding this, our intuition is that having data retrieved during the same time-frame the probabilities of sharing a similar vocabulary[9] (in terms of trending-topic hashtags, mentions, etc.) are higher than when using external data. Therefore, an approach exploiting only data provided in the task could perform better than one using additional data. With reference to the unconstrained setting, we observed a drop in the performance. In spite of this, we ranked in the $2^{nd}$ position when only unconstrained systems were considered.

Concerning Task B, our approach showed worst performance than in Task A. The results of both submissions were quite different. Probably this is due to the amount of classifiers involved in each ensemble. Overall, all the teams participating in the shared task had a lower performance in Task B demonstrating the difficulty of such a task. It is important to highlight that the *taskB_ENS_vot_U2* submission ranked in the $3^{rd}$ position when only the unconstrained setting was considered.

## 4 Error Analysis

We analyze those instances that were misclassified by our submissions in Task A observing different kinds of errors:

- Tweets where the ironic sense highly depends on the context where they are produced. In the following example it is not possible to understand the ironic intention without having more information: *@LukeLPearson hmm... let me think about that*[10]
- Tweets containing terms often used in ironic instances, such as "really". This is a disadvantage of word-based methods where terms highly related to a particular class provoke misleading classifications when they appear in other classes. The following is an example of this: *I'm really excited for next semester*[11]
- Tweets containing several hashtags. Most of the time our methods predicted such instances as ironic being in reality non-ironic: *@NormanWalshUK Stunning work. #british #textiles #footwear #madeinbritain #not-a-nike-clone*

## 5 Conclusions

In this paper we describe our participation at SemEval-2018 Task 3. We propose an ensemble method including well-known techniques together with a novel approach based on the words in a tweet to identify the presence of irony. From the results, we observe that our approach obtained relatively good results considering its simplicity. As future work, it could be interesting to enhance the tailored lexicon by exploiting more data and other strategies for collecting words which are likely to be used for achieving an ironic sense. Moreover, considering different criteria to assign the votes in our approach is also matter of further experiments.

## Acknowledgments

This research was funded by CONACYT project FC-2016/2410. The work of Paolo Rosso has been funded by the SomEMBED TIN2015-71147-C2-1-P MINECO research project.

[9] We found that training and test data for the task share around fifty percent of the vocabulary.

[10] Predicted class: *non-ironic*, Real class: *ironic*.

[11] Predicted class: *ironic*, Real class: *non-ironic*.

## References


Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. 2014. Modelling Sarcasm in Twitter, a Novel Approach. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 50–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29.

E. Fersini, F. A. Pozzi, and E. Messina. 2015. Detecting Irony and Sarcasm in Microblogs: The Role of Expressive Signals and Ensemble Classifiers. In *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–8.

Aniruddha Ghosh, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, John Barnden, and Antonio Reyes. 2015. SemEval-2015 Task 11: Sentiment Analysis of Figurative Language in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 470–478, Denver, Colorado. Association for Computational Linguistics.

Aniruddha Ghosh and Tony Veale. 2016. Fracking Sarcasm using Neural Network. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 161–169, San Diego, California. Association for Computational Linguistics.

Rachel Giora and Ofer Fein. 1999. Irony: Context and Salience. *Metaphor and Symbol*, 14(4):241–257.

H. P. Grice. 1975. Logic and Conversation. In P. Cole and J. L. Morgan, editors, *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.*, 11(1):10–18.

Delia Irazú Hernández Farías, Viviana Patti, and Paolo Rosso. 2016. Irony Detection in Twitter: The Role of Affective Content. *ACM Trans. Internet Technol.*, 16(3):19:1–19:24.

Delia Irazú Hernández Farías and Paolo Rosso. 2016. Irony, Sarcasm, and Sentiment Analysis. Chapter 7. In Federico Pozzi, Elisabetta Fersini, Enza Messina, and Bing Liu, editors, *Sentiment Analysis in Social Networks*, pages 113–127. Morgan Kaufmann.

Aditya Joshi, Pushpak Bhattacharyya, and Mark J. Carman. 2017. Automatic Sarcasm Detection: A Survey. *ACM Comput. Surv.*, 50(5):73:1–73:22.

Aditya Joshi, Vaibhav Tripathi, Kevin Patel, Pushpak Bhattacharyya, and Mark James Carman. 2016. Are Word Embedding-based Features Useful for Sarcasm Detection? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1006–1011.

Jihen Karoui, Farah Benamara, Veronique Moriceau, Viviana Patti, Cristina Bosco, and Nathalie Aussenac-Gilles. 2017. Exploring the Impact of Pragmatic Phenomena on Irony Detection in Tweets: A Multilingual Corpus Study. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, Valencia, Spain.

Peng Liu, Wei Chen, Gaoyan Ou, Tengjiao Wang, Dongqing Yang, and Kai Lei. 2014. Sarcasm Detection in Social Media Based on Imbalanced Classification. In *Proceedings of the Web-Age Information Management: 15th International Conference*, pages 459–471, Macau, China. Springer International Publishing.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 3111–3119.

Saif M. Mohammad, Xiaodan Zhu, Svetlana Kiritchenko, and Joel Martin. 2015. Sentiment, Emotion, Purpose, and Style in Electoral Tweets. *Information Processing & Management*, 51(4):480 – 499.

Debora Nozza, Elisabetta Fersini, and Enza Messina. 2016. Unsupervised Irony Detection: A Probabilistic Model with Word Embeddings. In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, pages 68–76.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, and Prateek Vij. 2016. A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks. *CoRR*, abs/1610.08815.

Tomáš Ptáček, Ivan Habernal, and Jun Hong. 2014. Sarcasm Detection on Czech and English Twitter. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 213–223, Dublin, Ireland. Association for Computational Linguistics.

Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 2015. Sarcasm Detection on Twitter: A Behavioral Modeling Approach. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, pages 97–106.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A Multidimensional Approach for Detecting Irony in Twitter. *Language Resources and Evaluation*, 47(1):239–268.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as Contrast between a Positive Sentiment and Negative Situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–714, Seattle, Washington, USA. Association for Computational Linguistics.

G. Salton, A. Wong, and C. S. Yang. 1975. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620.

Emilio Sulis, Delia Irazú Hernández Farías, Paolo Rosso, Viviana Patti, and Giancarlo Ruffo. 2016. Figurative Messages and Affect in Twitter: Differences between #irony, #sarcasm and #not. *Knowledge-Based Systems*, 108:132–143.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. SemEval-2018 Task 3: Irony Detection in English Tweets. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, SemEval-2018, New Orleans, LA, USA. Association for Computational Linguistics.