Irony detection in tweets: X2Check at Ironita 2018 (Short Paper)

Emanuele Di Rosa
Chief Technology Officer
App2Check s.r.l.
emanuele.dirosa
@app2check.com

Alberto Durante
Research Scientist
App2Check s.r.l.
alberto.durante
@app2check.com

Abstract

English. In this paper we describe and show the results of the two systems that we have specifically developed to participate at Ironita 2018 for the irony detection task. We scored as the third team in the official ranking of the competition, thanks to the X2C-B system, at a distance of just 0.027 of F1 score from the best system.

Italiano. *In questo report descriviamo i due sistemi che abbiamo sviluppato ad hoc per partecipare ad Ironita 2018, nello specifico al task di irony detection. Il nostro team è risultato essere il terzo classificato nella classifica ufficiale della competizione, grazie al nostro sistema X2C-B, che ha ottenuto un F1 score solo 0.027 inferiore rispetto al primo classificato.*

1 Introduction

In social media, the use of irony in tweets and Facebook posts is widely spread and makes very difficult for sentiment analysis tools to properly automatically classify people opinion (Hernández and Rosso, 2016). The ability to detect irony with high accuracy would bring an important contribution in opinion mining systems and lead to many industrial applications. For this reason, irony detection has been largely studied in recent research papers like (Farías et al., 2011), (Barbieri et al., 2014), (Farías et al., 2016), (Freitas et al., 2014).

In this paper we describe and show the results of the two systems that we have specifically developed to participate at Ironita 2018 (Cignarella et al., 2018) for the irony detection task. We scored as the third team in the official ranking of the competition, thanks to the X2C-B system, at a distance of just 0.027 of F1 score from the best system.

This paper is structured as follow: after the introduction we present the descriptions of our two systems submitted for the irony detection task; then we show and discuss the results on the official test set of the competition, finally we provide our conclusions.

2 Systems description

The dataset provided by Ironita organizers has been split into training set (80% of the documents) and development set (the remaining 20%). We randomly sampled the examples for each category, thus obtaining different sets for training/test set, by keeping the distribution of ironic and non-ironic samples through the two sets. We submitted two runs, as the results of the two different systems we developed for each category, called X2C-A and X2C-B. The former has been developed on top of the Scikit-learn library in Python language (Pedregosa et al., 2011), and the latter on top of the WEKA library (Frank et al., 2016) in JAVA language. In both cases, input text has been cleaned with a typical NLP pipeline, involving punctuation (with the exclusion of question/exclamation mark), numbers and stopwords removal. In particular, since it is still hard to detect irony in a text, very often also for humans, we tried to take advantage of features trying to help triggering the presence of irony. For instance, question and exclamation marks, text strings representing laughs, emoticons, mixed sentiment in the same sentence are some of the text features that we extracted from the text and represented with a specific explicit marker highlighting their presence.

Both the X2C-A and X2C-B unconstrained run were trained using the SENTIPOLC 2016 Irony training set and test set (Barbieri et al., 2016) as external source, in addition to the Ironita training set.

2.1 X2C-A

The X2C-A system has been created by applying an NLP pipeline including a vectorization of the collection of reviews to a matrix of token counts of bi-grams; then, the count matrix has been transformed to a normalized tf-idf representation (term-frequency times inverse document-frequency). For the training, we created an ensemble model, more specifically a voting ensemble, that takes into account three different algorithms: LinearSVC (an implementation of Support Vector Machines), Multinomial Naive Bayes and the SGD classifier. All of them have an implementation available in the Scikit-learn library. The ensemble model has been the best model in our model selection activity. In order to properly select the best hyper-parameters, we applied a grid search approach for each of the model in the voting ensemble. The resulting ensemble model showed a macro F1 score of 70.98 on our development set and is very close to the final result on the competition test set (shown in table).

	Acc	F1 ironic	Macro F1
LinearSVM	0.706	0.699	0.706
NB	0.706	0.699	0.706
SGD	0.697	**0.728**	0.693
Ensemble	**0.710**	0.709	**0.710**

Table 1: Results on the development set for X2C-A constrained.

2.2 X2C-B

In the model selection process, the two best algorithms have been Naive Bayes Multinomial and SMO, both using unigram features. We took into account the F1 score on the positive labels and the Macro-F1 in order to select the best algorithm. As shown in Table 2, Naive Bayes Multinomial reached a Macro F1 score 2.38% higher on the constrained run and a 14.2% on the unconstrained run, thus both the constrained and the unconstrained submitted runs were produced using this algorithm.

Comparing the results in Table 2 with the ones in Table 1, we can notice that X2C-B unconstrained reached the highest performance on the development set, while X2C-B constrained obtained the lowest score.

	F1 non-iro	F1 iro	Macro F1
NB-const	0.715	0.696	0.707
NB-uncon	**0.729**	**0.750**	**0.740**
SMO-const	0.678	0.689	0.683
SMO-uncon	0.704	0.492	0.598

Table 2: Results on development set for X2C-B.

3 Results and discussion

In Table 3 we show the results of our runs on the official test set of the competition. In accordance with what we noticed before, comparing Table 1 and Table 2, our best run is X2C-B unconstrained, which reached the best F1 overall on non-ironic documents; it also ranks fifth in the overall F1-score, at a distance of 0.027 from the best system. The performance of the X2C-A run is very similar to the unconstrained run, obtaining a F1-score that is only 0.002 higher than the constrained run. The difference between the two X2C-B runs is larger in relative terms, but is only of 0.021. We can also see that our X2C-B-u shows the best F1 score on the non-ironic tweets compared to all of the systems.

We added to this ranking also the model that reached the first position on the Irony task at SENTIPOLC 2016 (Di Rosa and Durante, 2016). The score of that model on this test set, called X2C2016 in the table, reached a F1-score of just 0.432, which is lower than the baseline of this year. This surprising result may indicate either that the irony detection systems had a great improvement in the past two years, or that irony detectors have a performance that is very much dependent on the topics treated in the training set, i.e. they are still not so good to generalize.

4 Conclusions

In this paper we described the two systems that we built and submitted for the Ironita 2018 competition for the irony detection task. The results show that our system X2C-B scored as the third team at a distance of just 0.027 of F1 score from the best system.

References

Alessandra Teresa Cignarella and Simona Frenda and Valerio Basile and Cristina Bosco and Viviana Patti and Paolo Rosso. 2018. *Overview of the Evalita 2018 Task on Irony Detection in Italian Tweets*

	team	F1 non-iro	F1 iro	F1
1	team 1	0.707	**0.754**	**0.731**
2	team 1	0.693	0.733	0.713
3	team 2	0.689	0.730	0.710
4	team 2	0.689	0.730	0.710
5	**X2C-B-u**	**0.708**	0.700	0.704
6	team 4	0.662	0.739	0.700
7	team 4	0.668	0.733	0.700
8	**X2C-A-u**	0.700	0.689	0.695
9	team 5	0.668	0.722	0.695
10	**X2C-A-c**	0.679	0.708	0.693
11	**X2C-B-c**	0.674	0.693	0.683
12	team 6	0.603	0.700	0.651
13	team 6	0.626	0.665	0.646
14	team 6	0.579	0.678	0.629
15	team 6	0.652	0.577	0.614
16	*baseline-1*	*0.503*	*0.506*	*0.505*
17	team 7	0.651	0.289	0.470
18	**X2C2016**	0.665	0.198	0.432
19	team 7	0.645	0.195	0.420
20	*baseline-2*	*0.668*	*0*	*0.334*

Table 3: Ironita 2018 official ranking.

(IronITA) in Proceedings of the 6th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA'18).

Francesco Barbieri and Valerio Basile and Danilo Croce and Malvina Nissim and Nicole Novielli and Viviana Patti. 2016. *Overview of the Evalita 2016 SENTIment POLarity Classification Task* in Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016), Napoli, Italy, December 5-7, 2016.

Emanuele Di Rosa and Alberto Durante. 2016. *Tweet2Check evaluation at Evalita Sentipolc 2016* in Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016), Napoli, Italy, December 5-7, 2016.

Eibe Frank, Mark A. Hall, and Ian H. Witten. 2016. The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition, 2016.

Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V. and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P. and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E. 2011. *Scikit-learn: Machine Learning in Python* in Journal of Machine Learning Research, pp. 2825–2830.

Farías, Delia Irazú Hernández et al. *Irony Detection in Twitter: The Role of Affective Content.* 2011. in ACM Trans. Internet Techn. 16 (2016): 19:1-19:24.

Barbieri, Francesco and Horacio Saggion. 2014. *Modelling Irony in Twitter: Feature Analysis and Evaluation.* in LREC (2014).

Delia Irazú Hernández Farías, Viviana Patti, and Paolo Rosso. 2016. *Irony Detection in Twitter: The Role of Affective Content.* in ACM Transaction Internet Technology 16, 3, Article 19 (July 2016), pp. 1-24. DOI: https://doi.org/10.1145/2930663

Freitas, Larissa and Vanin, Aline and Hogetop, Denise and N. Bochernitsan, Marco and Vieira, Renata. 2014. *Pathways for irony detection in tweets.* in Proceedings of the ACM Symposium on Applied Computing. 10.1145/2554850.2555048.

Hernández I., Rosso P. 2016. *Irony, Sarcasm, and Sentiment Analysis.* Chapter 7 In: Sentiment Analysis in Social Networks, F.A. Pozzi, E. Fersini, E. Messina, and B. Liu (Eds.), Elsevier Science and Technology, pp. 113-128

Sulis E., Hernández I., Rosso P., Patti V., Ruffo G. 2016. *Figurative Messages and Affect in Twitter: Differences Between #irony, #sarcasm and #not.* In: Knowledge-Based Systems, vol. 108, pp. 132–143