UniNE at PAN-CLEF 2022: Profiling Irony and Stereotype Spreaders on Twitter

(Notebook for PAN at CLEF 2022)

Catherine Ikae[1]

[1]*University of Neuchâtel, Switzerland, Avenue du 1er-Mars 26, 2000 Neuchâtel, Switzerland*

Abstract

This work proposes to solve the problem of profiling irony and stereotype spreaders on twitter using a random forest model with features obtained using chi square feature scoring. The task is to determine whether the author of a given Twitter feed in English spreads irony and stereotypes. The training sample contains timelines of authors sharing irony and stereotypes towards, for instance, women or the LGTB community. Transforming this question into binary classification problem which requires us to classify authors as ironic or not. Evaluation with 300 chi2 features shows an overall performance of ACC = 0.9722.

Keywords

Irony detection, machine learning, natural language processing, Random forest (RF), Twitter

1. Introduction

Irony uses metaphorical and sophisticated language to indicate the polar opposite of what is literally said. The purpose of sarcasm, a more aggressive form of irony, is to mock or humiliate a victim while not ruling out the possibility of harm. Stereotypes are frequently employed, particularly when discussing contentious issues such as immigration or sexism and misogyny.

Irony and stereotype has the unique virtue of making it difficult to bridge the gap between its literal and intended meaning. Detecting irony and stereotype behavior in online social networks

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy
catherine.ikae@unine.ch (C. Ikae)

like Facebook, Twitter, Instagram, polls, and other places has become a crucial task since it affects social and personal connections. Irony detection is a critical processing challenge in natural language processing (NLP), which is required for better understanding and to serve as an interface for machine-human communication.

Ironic and stereotyped tweets made in a friendly tone go unnoticed. People can use politeness to be sarcastic, using very formal language that do not fit the casual discourse. Irony is often shown by complimenting someone in a formal manner. Ironic and stereotyped tweets, by their very nature, lead to incorrect replies from review summarization systems, sentiment classifiers, review ranking systems, and any other application that deals with the semantics and pragmatics of text. Nonetheless, most social media and microblogging sites, such as Twitter, set text length limits that arguably do little to prohibit the use of innovative language such as sarcasm and irony, which allow strong views to be expressed effectively [1].

Irony detection was first explored by Reyes [2] [3], he approaches irony as a language phenomena, undistinguished from sarcasm, that incorporates something unexpected or contradictory, using decision trees to classify features indicating unexpectedness such as emotive words, contradictory phrases, and punctuation.

[4] is an example of automatic humor recognition in which he detected sarcasm in areas including politics, education, and humor. The hashtags #sarcasm, #politics, #education, and #humour were used to collect data from Twitter. He uses the American National Corpus Frequency Data and the morphology of tweets to create a measure of unexpectedness and probable ambiguities based on words that are primarily used in spoken language. Random forests and decision trees are used to classify the data.

To detect irony, [5] employed Logistic Regression and focused on the unexpectedness factor, which is defined as an emotional imbalance between words in a text. While [6] suggested a pattern-based technique for detecting irony that uses n-grams and combinations of adverbs and acronyms that imply comedy as characteristics. [7] introduced a new method that treats sarcasm as a binary classification job, pitting positive-negative statements against each other. WordNet-Affect and LWIC are used to extract lexical features, as well as so-called "pragmatic factors," which indicate user sentiment.

2. Corpus

The training corpus was available in the English to be used to determine determine whether its author spreads irony and stereotypes. The dataset had 210 documents of label NI (contains set of tweets)) and 210 documents of label I (tweets containing some form of irony). In our point of view, the problem is therefore to identify a set of tweets containing irony, leading to consider that the user generates and/or spreads stereotype [8][9]. This task will be performed only English language.

Table 1
Overall statistics about the training data

	irony(I)	not irony (NI)
Nb. doc.	210	210
Nb tweets	42000	42000
Mean length	5979	5296
|Voc|	57247	46642

Table 2
Sample of three tweets in English for each class

	English tweets
Class I	#user# $80 billion tshirt. Gotta be some #HASHTAG# in the fact that it actually physically exists.. Assuming it's not photoshopped.. #user# #user# #user# Kinda like rewards points from the merchant? That's not bad.. Fees? Converting to crypto is still extra effort and expense; electronic payment apps are just so easy and secure.. Outside the U.S., electronic payment has been ubiquitous for almost 3 decades. That's a big hill. #user# #user# #user# #user# You haven't explained shit. Scarcity has nothing to do with anything. You don't understand money. If you think "banks" print up billions of 1 dollar bills and just hand them out, you are an infant. Money is for using, not holding. There is no economy without spending. Get a job.
Class NI	Busy with plea deal in corruption case, Netanyahu is absent from the Knesset: Opposition leader Benjamin Netanyahu tells attorneys to push on in negotiations with Israel's attorney general. Sources close to the former PM claim he missed Knesset… #URL# Haaretz #URL# Prof. Mevorach warns: 'We don't know post-corona effects of Omicron': 'Children are a vulnerable population, we don't know post-corona effects of Omicron,' head of Hadassah Medical Center's COVID-19 ward warns. #URL# ArutzSheva #URL# The Iraqi cleric's gamble to sideline in Iran-backed factions in new government: The movement led by Shi'ite cleric Moqtada al-Sadr already re-elected a parliamentary speaker opposed by the Iran-aligned camp, and could leave them in Iraq's… #URL# Haaretz #URL#

As one can see in Tables 2, tweets in Category #I describe questions to a facts " *in the fact that it actually physically exists*" followed by ironic statements "*Assuming it's not photoshopped*". In tweets appearing under the second label, #NI the statements used in the tweets are facts followed by an explanation to the point noted.

3. Feature Selection

The process of picking a subset of the terms in the training set and using only this subset as features in text classification is known as feature selection. The objective of feature selection is twofold. First, it reduces the quantity of the effective vocabulary, making training and using a classifier more efficient. This is especially important for classifiers that are more expensive to train. Second, by removing noisy features, feature selection frequently improves classification accuracy. A noise feature is one that raises the classification error on new(unseen) data as it is added to the document representation.

Feature selection can be thought of as a way of substituting a complex classifier (one that uses all features) with a simpler one (using a subset of the features). Tokens are considered according to their document frequency (df) and word frequency difference in the two-stage feature selection technique. This exercise was completed with a three-fold threshold (df > 3) and a one-fold threshold (tf > 1). We design a feature set capable of differentiating each category using these two limitations. A term frequency difference is computed from the reduced number of tokens obtained by applying df > 3, but only tokens with a term frequency greater than 1 (tf > 1) are taken into account, leaving out tokens that appear just once in the text.

Chi-square (χ^2) is a function used to test the independence of two variables in the correct case, the independence of a feature and a category is provided by a χ^2 value. The higher value of the χ^2, the closer relationship the variables have [10]

$$\begin{aligned}\chi^2(t_i, c_j) &= \frac{n * ((p(t_i, c_j) * p(\bar{t}_i, \bar{c}_j))(p(t_i, \bar{c}_j) * p(\bar{t}_i, c_j)))^2}{p(t_i) * p(\bar{t}_i) * p(c_j) * p(\bar{c}_j)} \\ &= \frac{n * (a * d - c * b)^2}{(a + c) * (b + d) * (a + b) * (c + d)}\end{aligned} \tag{1}$$

Pointwise Mutual Information (PMI) measures how much information a term contains about a class. It measures how much information the presence/absence of a term contributes to making the correct classification decision. The magnitude of PMI will indicate if an association between a feature and a category exist or not [11] [10]. PMI can therefore be used to decide if a feature is informative or not, and a feature selection is done on that basis. Having less features often improves the performance of your classification algorithm. To calculate PMI, as a ratio the joint probability ($p(t_i, c_j)$) and the probability of occurrence of term t_i multiplied by the probability of selecting a text belonging to the category c_j.

$$\begin{aligned}PMI(t_i, c_j) = log_2(\frac{p(t_i, c_j)}{p(t_i) * p(c_j)}) &= log_2\left(\frac{\frac{a}{n}}{\frac{a+b}{n} * \frac{a+c}{n}}\right) \\ &= log_2\left(\frac{a * n}{(a + b) * (a + c)}\right)\end{aligned} \tag{2}$$

We discovered that such a figure is still too huge after evaluating the terms found in feature sets with thousands of terms. Only the top m terms (e.g., m = 300) depicting the highest discriminating powers selected with the chi2 (χ^2) and PMI feature selection techniques will be used in the model. The model with the best score was the chosen to build the final classifier.

Splitting the training data into train (300) and development (120), the two step feature selection reduces the features as shown in the Table 3.

Table 3
Two-step feature selection

	All	tf > 1	tf diff and df > 3
vocubulary I	46437	19316	5455
vocubulary NI	38379	16160	4322
Total number of features	84816	35476	9777

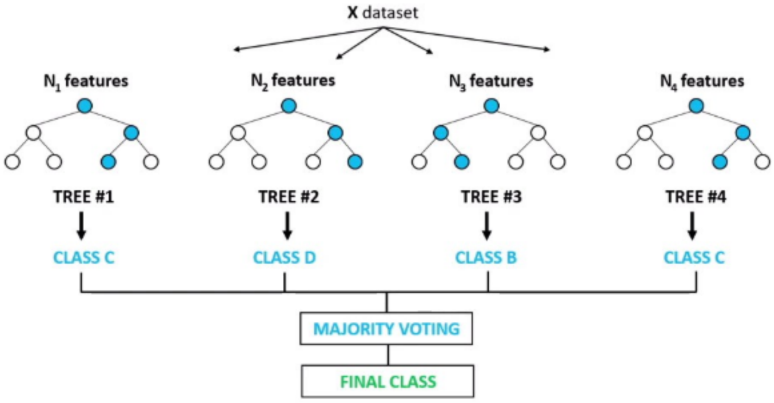
4. Random Forest Classifier

Random Forest consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model prediction [12]. This model can perform well given large feature sets because it combines the predictions of various decision trees to build a more robust classifier. While constructing new decision trees, this method uses a random subset of features which gets rid of spurious features and improving the robustness of our estimate.

The random forest method operates by following the steps below: 1): From the dataset provided, the algorithm selects random samples. 2): For each sample chosen, the algorithm will generate a decision tree. Then, for each decision tree constructed, it will acquire a prediction result. 3): For each expected outcome, voting will be conducted. It will employ mode to solve a classification problem and mean to solve a regression problem. 4): Finally, the algorithm will choose the prediction with the most votes as the final prediction.

In random forests, hyperparameters are used to either improve the model's performance and predictive capacity or to make it faster. The hyperparameters are listed as follows. 1). n estimators – the number of trees built by the algorithm before averaging the predictions. 2). max features – the number of features that a random forest evaluates while splitting a node. 3.) mini sample leaf – specifies how many leaves are necessary to separate an internal node. 4). n jobs– this parameter tells the engine how many processors it can use. It can only use one processor if the value is 1, but there is no limit if the value is -1. 5). random state– regulates the sample's unpredictability. If the model has a definite value of random state and is given the same hyperparameters and training data, it will always generate the same results. 6). oob score

Figure 1: Random Forest Classifier



– OOB is an abbreviation for "out of the bag." It's a cross-validation method based on random forests. In this case, one-third of the sample is used to evaluate the data rather than train it. These samples are drawn from a bag of samples.

Random forest has been used in a variety of applications, for example [13] combined data enrichment with the introduction of semantics in random forest to improve short text classification. The authors in [14] described a new method on random forest and feature selection (FS) for text classification and achieved macro-F1 score 73%. [15] performs sentiment classification of You Tube comments using the random forest, and Word2Vec Skip-gram for features extraction. [16] explores random forest with several term weighting method for sentiment analysis in Indonesian language.

5. Evaluation

To train our model, features are extracted from the training documents by taking into account the steps explained in section 3. Features to be considered must have a tf>1 and df>3 from which the ranking is done according to chi2 and PMI with k feature set of 300.

The accuracy of several classifiers are computed as shown in the Table 4. It was easy to analyse the performance of the classifiers where we can see that accuracy of XGB = 0.917 is the highest when all selected features are used. When chi2 was used to rank the features and using the top 300, random forest scored highest accuracy of RF = 0.942. Ranking the features again with PMI and using the top 300, again random forest had the best performance, RF = 0.925.

The use of RF in the final model with chi2 ranking was based on its overall highest score of 0.942 accuracy. The model was used to evaluated the test set which resulted into accuracy of 0.9722 as seen in Table 5.

Table 4
Evaluation based on different feature sizes

	features		
Classifiers	fs	chi2, k= 300	pmi, k= 300
KN	0.692	0.725	0.775
SVC	0.742	0.742	0.742
Extra Trees	0.858	0.892	0.867
Decision Tree	0.800	0.825	0.783
GaussianNB	0.858	0.875	0.808
BernoulliNB	0.858	0.833	0.850
MultinomialNB	0.558	0.558	0.542
MLP	0.867	0.850	0.758
SGD	0.625	0.633	0.625
LDA	0.833	0.625	0.500
Random Forest	0.858	**0.942**	**0.925**
AdaBoost	0.892	0.892	0.875
Bagging	0.833	0.875	0.883
Gradient Boosting	0.892	0.917	0.900
XGB	**0.917**	0.925	0.917
Logistic Regression	0.617	0.617	0.625

The final evaluation result is obtained on the TIRA platform [17] is exposed in Table 5.

Table 5
Official Evaluation with (m = 300)

Classifiers	Acc
Random Forest (Early Bird)	0.9722
Random Forest (Final)	0.9722

6. Conclusion

The research proposes a machine learning approach for detecting irony and stereotype spreaders. A random forest classifier was proposed. For the test set on TIRA, the final performance provided us an accuracy of roughly 0.9722. Because the features used in the classification are drawn in similar amounts from both classes, our approach is quite competent of identifying irony/non-irony tweets. The discriminating strength of each feature in the class is measured using a concept called probability difference. These characteristics highlight the distinction between the two classes, which are subsequently ranked based on their chi2 values to enable further reduction in the features.

References

[1] A. Ghosh, G. Li, T. Veale, P. Rosso, E. Shutova, J. Barnden, A. Reyes, SemEval-2015 task 11: Sentiment analysis of figurative language in Twitter, in: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Association for Computational Linguistics, Denver, Colorado, 2015, pp. 470–478. URL: https://aclanthology.org/S15-2080. doi:10.18653/v1/S15-2080.

[2] A. Reyes, P. Rosso, D. Buscaldi, From humor recognition to irony detection: The figurative language of social media, Data Knowledge Engineering 74 (2012) 1–12. doi:10.1016/j.datak.2012.02.005.

[3] A. Reyes, P. Rosso, T. Veale, A multidimensional approach for detecting irony in twitter, Language Resources and Evaluation 47 (2013) 239–268.

[4] F. Barbieri, H. Saggion, Modelling irony in twitter, 2014, pp. 56–64. doi:10.3115/v1/E14-3007.

[5] K. Buschmeier, P. Cimiano, R. Klinger, An impact analysis of features in a classification approach to irony detection in product reviews, in: Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Association for Computational Linguistics, Baltimore, Maryland, 2014, pp. 42–49. URL: https://aclanthology.org/W14-2608. doi:10.3115/v1/W14-2608.

[6] P. Carvalho, L. Sarmento, M. J. Silva, E. 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, TSA '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 53–56. URL: https://doi.org/10.1145/1651461.1651471. doi:10.1145/1651461.1651471.

[7] R. González-Ibáñez, S. Muresan, 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, Association for Computational Linguistics, Portland, Oregon, USA, 2011, pp. 581–586. URL: https://aclanthology.org/P11-2102.

[8] J. Bevendorff, B. Chulvi, E. Fersini, A. Heini, M. Kestemont, K. Kredens, M. Mayerl, R. Ortega-Bueno, P. Pezik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle, Overview of PAN 2022: Authorship Verification, Profiling Irony and Stereotype Spreaders, and Style Change Detection, in: M. D. E. F. S. C. M. G. P. A. H. M. P. G. F. N. F. Alberto Barron-Cedeno, Giovanni Da San Martino (Ed.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022), volume 13390 of *Lecture Notes in Computer Science*, Springer, 2022.

[9] O.-B. Reynier, C. Berta, R. Francisco, R. Paolo, F. Elisabetta, Profiling Irony and Stereotype Spreaders on Twitter (IROSTEREO) at PAN 2022, in: CLEF 2022 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2022.

[10] J. Savoy, Comparative evaluation of term selection functions for authorship attribution, Digital Scholarship in the Humanities 30 (2013) 246–261. URL: https://doi.org/10.1093/llc/fqt047. doi:10.1093/llc/fqt047.

[11] K. Church, P. Hanks, Word association norms, mutual information, and lexicography, Computational Linguistics 16 (2002). doi:10.3115/981623.981633.

[12] L. Breiman, Random forests–random features (2021).

[13] A. Bouaziz, C. Dartigues-Pallez, C. da Costa Pereira, F. Precioso, P. Lloret, Short text classification using semantic random forest, 2014, pp. 288–299. doi:10.1007/978-3-319-10160-6_26.

[14] S. Maruf, K. Javed, H. Babri, Improving text classification performance with random forests-based feature selection, Arabian Journal for Science and Engineering 41 (2015). doi:10.1007/s13369-015-1945-x.

[15] S. Khomsah, Sentiment analysis on youtube comments using word2vec and random forest, Telematika 18 (2021) 61. doi:10.31315/telematika.v18i1.4493.

[16] M. Fauzi, Random forest approach for sentiment analysis in indonesian language, Indonesian Journal of Electrical Engineering and Computer Science 12 (2018) 46–50. doi:10.11591/ijeecs.v12.i1.pp46-50.

[17] M. Potthast, T. Gollub, M. Wiegmann, B. Stein, TIRA Integrated Research Architecture, 2019, pp. 123–160. doi:10.1007/978-3-030-22948-1_5.