The Interplay between Emotions and Irony in Text Classification

Sofia Casadei
University of Stuttgart
st172313@stud.uni-stuttgart.de

Abstract

Irony is an intricate phenomenon that has the potential to undermine Natural Language Processing (NLP) systems for the classification of emotions, as it can mask the true sentiment conveyed by a text. This study analyses the effects of giving classifiers information regarding the presence of irony in Twitter data, revealing that this is not sufficient to improve performance. Similarly, irony detectors do not seem to benefit from the addition of information about the emotion of a tweet. Moreover, a feature analysis revealed that there may be a few soft linguistic indicators of irony. Finally, co-occurrence patterns between irony and both positive and negative emotions were found, providing additional evidence for the complexity of this phenomenon.

1 Introduction

Irony, despite being a widely used conversational tool, remains a challenging concept to exhaustively put into words. Cambridge Dictionary (n.d.) defines it as "the use of words that are the opposite of what you mean, as a way of being funny". For example, a sentence such as (1), if uttered by someone who has just split their coffee, would most likely mean 'this day started off badly'.

(1) "Great start of the day!"
(2) "I love children who keep their rooms tidy"
(3) "Hurray! I have been fired again!"

The idea that employing irony means saying the opposite of what is meant accords with the influential account of irony proposed by Grice (1975). However, this definition seems to only scratch the surface of this complex phenomenon, since there are instances, such as (2), that do not fit that description.

There exist several other accounts of irony, including the Echoic-Mention Theory proposed by Sperber and Wilson (1981), which defines irony in terms of its purpose: it implicitly mentions a previous proposition [1], and it conveys the speaker's (negative) attitude towards it.

Automatic irony detection is particularly arduous for NLP systems, as these tend to rely on the words presented and do not have the inherent ability to understand complex associations that require world knowledge and familiarity with the conversational background (Van Hee et al., 2016). On the other hand, computers are excellent at extracting patterns and correlations. If such an operational definition was found, it could lead to a better understanding of irony as a linguistic phenomenon (Buschmeier et al., 2014).

The presence of irony in corpora acts as a confounding influence for NLP systems whose aim is to capture the emotion of a text. This is because irony often involves a mismatch between the literal meaning (and overt sentiment) and the intended meaning (and true sentiment), with one of the two being implicit (Van Hee et al., 2018a). For example, (3) would most likely be classified as expressing joy, even though it is clear to humans that it conveys a negative emotion.

By analysing irony computationally, I attempt to discover how it is realised linguistically and whether there are linguistic features that can be considered 'markers of irony'. This study focuses on emotion classification and aims at training NLP models that can successfully detect irony in text and classify text according to the emotion that it carries. Of most interest is whether the detection of irony in text can improve the performance of emotion classifiers, as well as whether information about the emotion of a text can improve the performance of irony detectors. Here, irony and sarcasm are not distinguished, due to the lack of definitive and agreed-upon difference between the two.

[1] A previous proposition may be an experience, utterance, or thought - of the speaker or of somebody else.

2 Methodology

This study tackles both the task of irony detection and emotion classification as supervised classification problems. Irony detection consists of binary classification where a text sample T is labeled with either 1 or 0, which translates to ironic or non-ironic, respectively.

$$T \rightarrow \{0, 1\}$$

Emotion classification is an instance of single-label multi-class classification, where a text sample T is mapped to one emotion e from a set of possible emotion labels.

$$T \rightarrow E : \{e_1, e_2, e_3, ..., e_n\}$$

The practical implementation is carried out in Python and employs ktrain (Maiya, 2020), an open-source wrapper for the deep learning library Keras (Chollet et al., 2015). More specifically, it uses the pre-trained HuggingFace (Wolf et al., 2020) transformer model RoBERTa (Liu et al., 2019). [2]

Throughout this study, the following datasets are used:

- For emotion classification, the "International Survey on Emotion Antecedents and Reactions" (ISEAR) dataset (Scherer and Wallbott, 1994), a collection of self-reported emotional events, labeled with one of seven possible emotions (joy, fear, anger, sadness, disgust, shame, and guilt).
- For emotion classification, a modified version of the dataset from SemEval 2018 Task 9: "Sentiment Analysis in Twitter" (Mohammad et al., 2018), published as part of the TweetEval benchmark (Barbieri et al., 2020). It consists of tweets, labeled with one of four possible emotions (joy, optimism, sadness, anger).
- For irony detection, a collection of tweets published as part of SemEval 2018 Task 3 Subtask A: binary "Irony Detection in English Tweets" (Van Hee et al., 2018b).

3 Experiments and Results

3.1 Preliminaries

In order to perform experiments, it was necessary to implement and fine-tune an emotion classifier and an irony detector, so that they could be loaded when needed as predictors. In the following, I briefly describe the findings that were observed while finding the best model, namely the effect of text preprocessing and that of emojis.

3.1.1 Emotion classifier

Results suggest that linguistic preprocessing of the ISEAR dataset does not improve emotion classification performance of a pre-trained RoBERTa transformer. In fact, the model trained on raw data showed better accuracy. More specifically, the preprocessing step involved the removal of (some) stop words, the expansion of contractions and the removal of punctuation. When experimenting with the TweetEval data, it was found that the presence of (decoded) emojis does not improve performance. Nevertheless, some preprocessing was performed, including the conversion of urls to the token 'url' and that of tags to the token 'tagged_user'. The best model was found to be RoBERTa after the complete removal of emojis. Accuracies are reported in Table 1 below.

3.1.2 Irony Detector

The RoBERTa model was fine-tuned on the SemEval irony dataset. The best model was found to be the one trained after the removal of emojis, providing an accuracy of 0.75 - which decreases by 1% when emojis are included.

Classifiers	Accuracy
EC_RoBERTa_ISEAR_preprocess	0.71
EC_RoBERTa_ISEAR_no-preprocess	0.72
EC_RoBERTa_TweetEval_emojis	0.80
EC_RoBERTa_TweetEval_no-emojis	**0.81**
ID_RoBERTa_SemEval_emojis	0.74
ID_RoBERTa_SemEval_no-emojis	**0.75**

Table 1: Classifiers for Emotion Classification (EC) and Irony Detection (ID).

3.2 Irony detection with emotion labels

The aim of this experiment was to test whether the addition of emotion labels to tweets results in higher irony detection performance. The assumption is that, if we have information about the emotion a text conveys, it may be easier to detect whether it is ironic or not. In addition, this might reveal co-occurrence patterns between emotions and irony.

Results show that the addition of emotion labels to a dataset consisting of ironic and non-ironic tweets does not improve irony detection

[2] The implementation of this study can be found at https://github.com/sofi444/CLTeamLab2021/tree/main/Phase%202:%20advanced%20methods

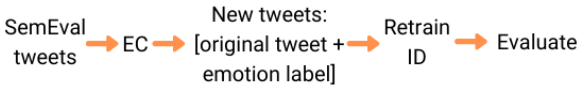

Figure 1: Procedure involved in experiment 1: each tweet from the irony dataset is passed to the emotion classifier (EC) which makes a prediction about the emotion of the tweet. The emotion label is then added to the end of the original text. Then, an irony detector (ID) is trained on the new dataset and evaluated against the original ID.

performance. However, an improvement was observed when using the emotion classifier trained on TweetEval data (with labels joy, optimism, anger, sadness) instead of the one trained on ISEAR data (with labels joy, fear, anger, sadness, disgust, shame, and guilt). Nevertheless, the irony detector trained on the original tweets outperformed both variations.

Models	Accuracy
ID_RoBERTa_SemEval_no-emojis	**0.75**
ID_RoBERTa_with-emotion_ISEAR	0.71
ID_RoBERTa_with-emotion_TE	0.73

Table 2: Results of experiment 1

3.3 Emotion classification with irony labels

This experiment is the counterpart of the preceding one, as it examines whether the addition of irony labels (<irony>, <not-irony>) improves emotion classification performance. The assumption that motivates this experiment is that information regarding the presence or absence of irony should facilitate the recognition of the emotion conveyed by a text. [3]

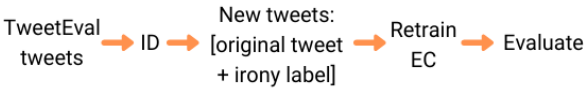

Figure 2: Procedure involved in experiment n.2. Each tweet from the TweetEval dataset is passed to the irony detector (ID) which makes a prediction about whether the tweet is ironic or not. The irony label is then added to the end of the original text. Then, an emotion classifier (EC) is trained on the new dataset and evaluated against the original EC.

[3]Only the TweetEval dataset was used, following the assumption that some irony will be present, because of the nature of social media text. On the other hand, the ISEAR dataset (consisting of self-reported emotional events is very unlikely to contain ironic statements.

Models	Accuracy
EC_RoBERTa_TweetEval_no-emojis	**0.81**
EC_RoBERTa_with-irony_TE	0.79

Table 3: Results of experiment 2

It was found that the addition of irony labels does not improve accuracy for the task of emotion classification of tweets. In fact, the original emotion classifier trained on the original data provided a performance that was 2% higher.

3.4 Feature analysis for irony detection

The aim of this experiment was to analyse whether there exist features that are good predictors of irony in text. Tweets were tested for the presence of each of the features listed in table 4, and the relationships between these features and the irony labels were examined.

Features	All tweets(%)	Irony tweets(%)
all caps	0.63	23.06
ellipsis	11.88	46.90
pol. change	14.02	55.03
laugh	3.77	53.73
emojis	9.00	42.89
interjection	1.10	63.64
exclamation	17.27	51.56
hard excl.	4.34	54.97
interrogative	9.16	42.48
hard interr.	0.26	45.45
user tag	33.65	42.27

Table 4: The relationships between features and the presence of irony. The 2nd column holds the percentage of ironic tweets with a given feature out of all the tweets in the dataset, while the 3rd column holds the percentage of ironic tweets that contain a given feature out of all the ironic tweets. The features are binary, meaning they evaluate to either 1 if they are present in the tweet, or 0 if they are not; feature definitions are provided in the Appendix.

No definitive and strong indicator of irony was found. When looking at the whole dataset, many of the probabilities are close to 50%, meaning the distribution of irony and not-irony is nearly equal when the given feature occurs. However, a few findings stood out. 34% of the ironic tweets contain at least one tagged user and 17% contain at least one exclamation mark. Interestingly, only 1% of the ironic tweets contain an interjection, but 64% of all tweets in the dataset which contain an interjection are ironic. Polarity change was found in 14% of the ironic tweets, and the probability of a tweet being ironic when this feature is present is slightly higher than that of being non-ironic.

3.5 Irony-emotion co-occurrence patterns

This experiment's goal is to find out which emotion (if any) is more likely to be conveyed by ironic tweets. Both emotion classifiers (the one trained on ISEAR data and the one trained on TweetEval data) were employed, meaning that two different sets of labels were tested.

It was observed that, according to both label sets, the prevalent emotion of ironic tweets is 'joy' and the second emotion most likely to be conveyed with ironic statements is 'anger'.

ISEAR	%	TweetEval	%
I + joy	29.51	I + optimism	14.39
I + sadness	6.54	I + joy	52.38
I + fear	11.77	I + anger	22.03
I + disgust	9.05	I + sadness	11.20
I + shame	14.29		
I + guilt	4.81		
I + anger	24.02		

Table 5: Percentages of ironic tweets that were labelled with the above emotions.

4 Discussion and conclusion

This study set out to investigate the interplay between emotions and irony. More specifically, the aims were: 1) to reveal whether the addition of irony labels to tweets results in improved performance for the task of multi-label emotion classification; 2) similarly, to analyse whether the addition of information regarding the emotion conveyed by a tweet improves performance for the task of irony classification; 3) to find out whether there are linguistic features that are often present in ironic tweets, and therefore can be considered good predictors of irony; 4) to discover co-occurrence patterns between irony and emotions.

Neither emotion classification nor irony detection showed an improved performance when the extra labels were added. This may be explained in terms of the quality of the predicted labels: neither predictor has near-perfect accuracy. For instance, the best irony detector has an accuracy of 0.75, meaning that 25% of the irony labels can be expected to be wrong. More specifically, it tends to often generate false positives (i.e., to predict irony when a tweet is actually not ironic). Nevertheless, this accuracy is relatively high for irony detection, with the winning implementation of SemEval 2018 achieving 0.73. Though, it should be noted that at the time of the competition, the RoBERTa model used in my implementation had not yet been published.

With regard to the linguistic realisation of irony, soft indicators of irony may be the presence of interjections (i.e., words such as 'uh', 'gosh', 'damn'), exclamation marks (especially if several are adjacent). Moreover, ironic tweets may be more likely to contain words with contrasting polarities.

Irony appears often with 'joy' and 'anger'. This seemingly contradicting pattern may be due to the fact that our definition of irony does not distinguish between irony and sarcasm. Sarcasm is thought to be different from irony in that it is more aggressive and its goal is to cause verbal harm (Dynel, 2014), while irony is a more innocent form of humour. In terms of the politeness framework popular in sociolinguistics, sarcasm constitutes a "face-threatening, attacking criticism [...] while irony is a face-saving [act]" (Barbe, 1995).

In addition, it was found that the emotion classifier trained on TweetEval data outperforms the one trained on ISEAR data. This was expected since the ISEAR data is labeled with 7 (vs. 4) emotions, of which only one is positive; the realm of negative emotions is much more fine grained, making it more difficult to distinguish between such similar and subjectively-defined emotions. Moreover, preprocessing of text and the inclusion of emojis in tweets worsened performance in both emotion classification and irony detection.

To conclude, irony labels are not helpful for emotion classification. Similarly, emotion labels do not improve performance in the (binary) irony detection task. Irony detection specifically is a challenging task because, on the linguistic level, no definitive markers of irony seem to be present. Irony is a multi-faceted tool which, depending on how it is used, can convey positive emotions or negative ones.

References

Katharina Barbe. 1995. Irony in context. Amsterdam/Philadelphia. John Benjamins Pub. Co.

Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. 2020. TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification. *arXiv:2010.12421 [cs]*. ArXiv: 2010.12421.

Konstantin Buschmeier, Philipp Cimiano, and Roman Klinger. 2014. 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*, pages 42–49, Baltimore, Maryland. Association for Computational Linguistics.

Francois Chollet et al. 2015. Keras.

Cambridge Dictionary. n.d. Irony. In *Cambridge Dictionary*.

Marta Dynel. 2014. Isn't it ironic? Defining the scope of humorous irony. *HUMOR*, 27.

Herbert Paul Grice. 1975. Logic and conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press, New York.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*. ArXiv: 1907.11692.

Arun S. Maiya. 2020. ktrain: A Low-Code Library for Augmented Machine Learning. *arXiv:2004.10703 [cs]*. ArXiv: 2004.10703.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. SemEval-2018 Task 1: Affect in Tweets. In *Proceedings sof The 12th International Workshop on Semantic Evaluation*, pages 1–17, New Orleans, Louisiana. Association for Computational Linguistics.

K. R. Scherer and H. G.. Wallbott. 1994. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of Personality and Social Psychology*, 66(2):310–328.

Dan Sperber and Deirdre Wilson. 1981. Irony and the use-mention distinction.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2016. Exploring the Realization of Irony in Twitter Data. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1794–1799, Portorož, Slovenia. European Language Resources Association (ELRA).

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018a. Exploring the fine-grained analysis and automatic detection of irony on Twitter. *Language Resources and Evaluation*, 52(3):707–731.

Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018b. SemEval-2018 Task 3: Irony Detection in English Tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv:1910.03771 [cs]*. ArXiv: 1910.03771.

4.1 Appendix

Feature definitions:

- All caps: all (or nearly all) words in a tweet are fully capitalised.
- Ellipsis: tweet contains ellipsis (at least 3 adjacent period marks).
- Polarity change: words contained in a tweet have contrasting polarities, meaning the polarity of words is not consistent throughout the sentence. Sentiment polarity values were collected using the function 'sentiment.polarity' of TextBlob [4].
- Laugh: tweet contains at least one laughing acronym (the accepted ones are 'haha' and 'lol' and variations of these).
- Emojis: tweet contains at least 1 emoji.
- Interjection: tweet contains at least one word that was tagged as interjection by the NLTK pos tagger [5].
- Exclamation: tweet contains at least one exclamation mark.
- Hard exclamation: tweet contains 2 or more adjacent exclamation marks.
- Interrogative: tweet contains at least one question mark.
- Hard interrogative: tweet contains 2 or more adjacent question marks.
- User tag: tweet contains at least one tagged user (@username).

[4]Loria, S. (2018). TextBlob: Simplified Text Processing — TextBlob 0.16.0 documentation. URL: https://textblob.readthedocs.io/en/dev/

[5]Bird, Steven, Edward Loper and Ewan Klein (2009), Natural Language Processing with Python. O'Reilly Media Inc.