Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 6976–6984
Marseille, 20-25 June 2022

Tackling Irony Detection using Ensemble Classifiers and Data Augmentation

Christoph Turban, Udo Kruschwitz
University of Regensburg
christoph.turban@gmail.com, udo.kruschwitz@ur.de

Abstract

Automatic approaches to irony detection have been of interest to the NLP community for a long time, yet, state-of-the-art approaches still fall way short of what one would consider a desirable performance. In part this is due to the inherent difficulty of the problem. However, in recent years ensembles of transformer-based approaches have emerged as a promising direction to push the state of the art forward in a wide range of NLP applications. A different, more recent, development is the automatic augmentation of training data. In this paper we will explore both these directions for the task of irony detection in social media. Using the common SemEval 2018 Task 3 benchmark collection we demonstrate that transformer models are well suited in ensemble classifiers for the task at hand. In the multi-class classification task we observe statistically significant improvements over strong baselines. For binary classification we achieve performance that is on par with state-of-the-art alternatives. The examined data augmentation strategies showed an effect, but are not decisive for good results.

Keywords: Irony detection, Ensembles, Transformers, BERT, Data augmentation

1. Motivation

Working with figurative language has long been considered one of the most difficult topics in Natural Language Processing (NLP), e.g. (Reyes et al., 2012). In contrast to literal language, this type of language uses stylistic devices such as metaphors, rhetorical questions as well as irony. For its understanding, a reader must use more abstract thinking and interpret information beyond syntax and semantics because the information needed for understanding is not expressed in the text itself (Reyes et al., 2012). If the underlying information is not unveiled, the actual meaning stays hidden and the speaker might be misunderstood. The detection of figurative language is particularly important in cases where irony is present because the actual meaning is often the opposite of the literal (Wilson and Sperber, 1992). The interpretation of irony can in fact be traced back all the way to Cicero's De Oratore.[1]

Take the expression *"Great game"* (assuming this has been said by a keen supporter of a sports team that has just lost a match). This is an example of irony where the speaker says something positive to express a negative sentiment. What makes this difficult is that the meaning is not expressed with adjectives like 'terrible', but instead, the polarity of the word 'great' must be reversed. In order to know that 'great' must not be interpreted in a literal way, the interpreter has to know the broader context.

One might argue that this is not much of a problem in social chit-chat, it turns out that the problem has much wider implications as irony is widespread in abusive content, and therefore sturdy detection of irony and sarcasm can help to recognize online toxicity and harassment (Van Hee, 2017). For example, Sanguinetti et al. (2018) revealed that 11% of hateful tweets in Italian are expressed through irony. Chatbots are also a target of highly emotional language which includes irony and sarcasm, to which it must be able to adequately respond. If it lacks this and other aspects of human language comprehension, interactions feel robotic and standardized which is not desirable for language depth and humanness (Croes and Antheunis, 2021). More broadly, irony and sarcasm can play a role in the wider context of growing problems such as cyber-aggression and cyber-bullying (Mladenović et al., 2022).

The wide-spread application of transformer models since 2018, (Devlin et al., 2019), and their superior performance over other algorithms makes it desirable to explore their full potential when applied to irony detection. This is especially important for irony detection with different types of irony present where systems are known to perform much worse (Van Hee et al., 2018). We would argue that beyond the simple binary classification it is this area that will become more interesting over time, because a real-world application which aims to detect ironic and sarcastic expressions should be able to differentiate between, for example, situational irony and insulting sarcasm.

This paper presents a contribution towards addressing irony detection using ensemble classifiers that tap into the power of transformer-based language models. Rather than simply applying these models we also address the commonly observed NLP problem of scarcity of training data. We explore the use of data augmentation methods and their implementation into different classifiers. This will provide guidance on how well newer transformer solutions can classify irony. To foster reproducibility we make all code and detailed experimental results available to the community

[1] `http://www.attalus.org/old/deoratore2D.html`

via GitHub[2]. The overriding research questions of our work are as follows:

- **RQ 1**: Is an ensemble classifier combining fine-tuned transformer models able to achieve better results in irony detection than state-of-the-art baselines?
- **RQ 2**: Does the use of data augmentation methods improve classification quality?

2. Related Work

2.1. Irony and Sarcasm

Irony has been studied in disciplines such as psychology (Clark and Gerrig, 1984), philosophy (Colebrook, 2002) and linguistics (Barbe, 1995), but no commonly accepted definition has emerged. Many agree that irony and sarcasm allow users to express themselves by using words in a creative and non-literal sense (Farías et al., 2016), and express affective contents like emotions and attitudes towards a target. That target can be a person, event or object (Farías et al., 2016). There has been an active debate around irony and sarcasm as similar linguistic phenomena as illustrated by Attardo (2007) who concluded that sarcasm represents an aggressive type of irony with a sharper tone which has also been underlined by Bowes and Katz (2011). There have also been other explorations into the specific characteristics of sarcasm distinguishing it from other forms of irony, e.g. (Wang, 2013; Sulis et al., 2016).

2.2. Automatic Irony Detection

The first approaches to computationally formalize irony for computers go way back in time. One of the first to attempt it was Utsumi in 1996. It was at first strongly related to humor recognition of one-liner jokes (Utsumi, 1996). Other studies followed with approaches explaining the cognitive process of the underlying irony. The first relevant successes next to more theoretical solutions which led to today's irony detection systems were presented about a decade ago. For example, Carvalho et al. (2009) used punctuation and quotation marks to detect irony in user-generated content. It followed a period where these hand-crafted features were more and more combined with machine learning methods which allowed the implementation of better-performing models. Rule-based approaches mimic the human mind and can achieve precise results, however, they are limited to the rules they are provided with which makes them brittle. Van Hee (2017) gives an overview of rule-based approaches which utilize different specific properties of a text.

More recently deep learning approaches have been applied to the task of irony detection where features are automatically derived from texts and thus eliminate the time-consuming extraction of rules, e.g. (Potamias et al., 2020; Lee et al., 2020). Many of these recent developments are based on the introduction of transformer-based approaches such as BERT which apply a two-step training phase of pre-training and fine-tuning (Devlin et al., 2019). Classical machine learning algorithms such as SVM, kNN or tree-based models and other probabilistic classification models have therefore been almost replaced since they have, among other shortcomings, a large demand for hand-crafted features to perform adequately.

While the most relevant related work here is around social media feeds posted in English it should be pointed out that there have been many other approaches that focus on different domains, e.g. (Cervone et al., 2017), and different languages, e.g. (González et al., 2019). In particular the choice of application domain has implications on what approach to take in detecting irony – while emojis and external links are key features of Twitter posts this is not the case in traditional news articles, for example.

Shared tasks in the research community such as *SemEval 2018 Task 3 Irony Detection in English Tweets* (Van Hee et al., 2018), and the *Second Workshop on Figurative Language Processing 2020 in Sarcasm detection* (Ghosh et al., 2020), as well as language-specific challenges such as *Irony Detection in Portuguese at IberLEF 2021* (Corrêa et al., 2021) have been organised and they allow a broad insight into state-of-the-art approaches.

2.3. Application Areas

Going back to the automatic identification of ironic content with the help of a computer program, multiple use cases can be identified. For example in the task of automatic identification of sentiment, the detection of ironic and sarcastic texts is important. A huge amount of sentiment information is available from online social networks like blogs and forums and micro-blogging platforms like Twitter since they are a rich source of user-generated content (Pak and Paroubek, 2010; Feldman, 2013). These user-generated texts contain a large amount of ironic or sarcastic expressions (Farías et al., 2016), which can act as sentiment shifters (Liu, 2020) and thus have a significant impact on the way they should be interpreted (Maynard and Bontcheva, 2015). If these texts are not classified as ironic, they lower the performance of such systems (Maynard and Greenwood, 2014; Rosenthal et al., 2014). Therefore, to achieve high precision in sentiment classification the efficient detection of sentiment shifting irony and sarcasm needs to be addressed.

2.4. Data Augmentation

There are different ways to effectively address the issue of data scarcity when it comes to preparing training data for supervised machine learning. Most recent ideas include the automatic generation of data using autoregressive models such as GPT-2 and GPT-3, e.g.

[2] https://github.com/ChristophTurban/LREC-Irony-Detection-Ensemble-Classifier

(Wullach et al., 2021; Whitfield, 2021). More linguistically motivated approaches have also been explored, e.g. (Juuti et al., 2020; Dekker and van der Goot, 2020). In particular the adoption of translating from one language into another appears to be appealing, e.g. (Tran and Kruschwitz, 2021). Data augmentation has actually already been shown to be useful in sarcasm detection when done correctly (Lee et al., 2020).

2.5. Datasets and Annotation Methods

Many datasets with ironic data exist. Corpora which provide annotation for ironic expressions stem from a broad range of sources and utilize different approaches for extraction and annotation. Most are in the form of a binary classification and do not distinguish between different types of irony and sarcasm.

Corpus Acquisition. Kreuz and Caucci (2007) used Google Book Search to find sarcastic expressions. They searched for the collocation "said sarcastically" and extracted the phrase which referred to it. Carvalho et al. (2009) used different linguistic patterns of oral or gestural clues. They identified ironic sentences by looking for emoticons, laughter expressions, sentence markers, and more. To collect ironic expressions from platforms like Twitter, low effort is needed. Users self-annotate tweets by using tags like #irony. Simply relying on this approach however means that some cases of irony and sarcasm where people did not find it necessary to append markers, might stay hidden. If these cases are not considered, applications might perform badly on unlabeled data. However, there is a body of work that aims at going beyond such simple heuristics, e.g. (Basile et al., 2021; Cignarella et al., 2018; Bueno et al., 2019).

Many datasets have in common that they only declare whether an expression is ironic or not, but they do not distinguish between different types of irony like in Van Hee's work (Van Hee et al., 2016). Similar work has been done by Karoui et al. (2017), where ironic expressions were categorized depending on the presence of different stylistic devices and irony markers.

Multi-class Irony Annotation. The annotation scheme for fine-grained types of irony deployed by *SemEval 2018 Task 3 , Irony Detection in English Tweets* (Van Hee et al., 2018) was that proposed by Van Hee (2017). The scheme distinguishes between "ironic by the means of a polarity clash", "situational irony", "other type of irony" and "not ironic". Motivated by the fact that categorical labels (e.g. #irony #sarcasm #not) are only sufficient for labeling data for a binary classification task, she proposes a manual annotation based on a set of newly developed coding principles (Van Hee et al., 2016). This was the first time that guidelines were proposed for finer labeling of ironic text which could differ between irony cases like irony by polarity clash or situational irony. Since humans can fail to detect irony, as for example González-Ibánez et al. (2011) has shown, two inter-annotator agreement studies were performed to assess the credibility of initially retrieved tweets, see Van Hee et al. (2018) for details. Results of these show that these judgements are very subjective and vary for each person and thus can rightfully be viewed critically.

2.5.1. SemEval2018 Task 3 Corpus

The common reference corpus for irony detection is the one proposed for SemEval2018 Task 3.[3] For its construction, Twitter was searched for ironic tweets using hashtags such as #irony, #sarcasm or #not. In total, 3000 tweets were collected. To make the dataset suitable for a classification task, additional steps were taken. More non-ironic tweets were added to balance the distribution between ironic and non-ironic tweets. Those were taken from the same set of tweets that contained the ironic tweets and manually labeled to be non-ironic. The irony marking hashtags were removed as well. This was applied to non-ironic tweets as well, which can lead to problems like the following. "#Myanmar #men #plead #not #guilty to #murder of #British #tourists. [...]" is an example of a non-ironic tweet. Removal of #not changed the meaning completely which could also have introduced bias. After the split of the dataset into train-set and test-set, tweets for which additional context would be required to understand them were removed from the test-set. This means that the train-set can contain cases where the irony is not recognizable because for example, the irony marking hashtag is missing.

3. Methodology

3.1. Corpus

The *SemEval 2018 Task 3* dataset is being used in this work because it is a commonly used reference benchmark. It also means that we will be able to compare our experimental results against the top-performing systems reported in the literature.

3.2. Ensemble Approach

Ensemble classifiers have been found to be the most competitive in a range of NLP tasks outperforming individual classifiers. As an illustration: a look at today's leaderboard of the SQuAD 2.0 question-answering challenge demonstrates that the top ten best-performing approaches (and beyond!) all use an ensemble approach.[4] This is by no means a very recent development. Hagen et al. (2015) managed to win SemEval2015's Task 10 Subtask B of sentiment detection (Rosenthal et al., 2015). Zimmerman et al. (2018) also observed a significant improvement of an ensemble model over individual models in the task of hate speech detection. Ensemble learning methods like boosting have also been proposed (see (Aggarwal and

[3] https://competitions.codalab.org/competitions/17468

[4] https://rajpurkar.github.io/SQuAD-explorer/ (accessed 15 Jan 2022)

Zhai, 2012)), where a classifier gets trained on different parts of data. This will not be used here since the dataset is relatively small and overfitting might happen. We use simple majority voting as illustrated in Fig. 1.

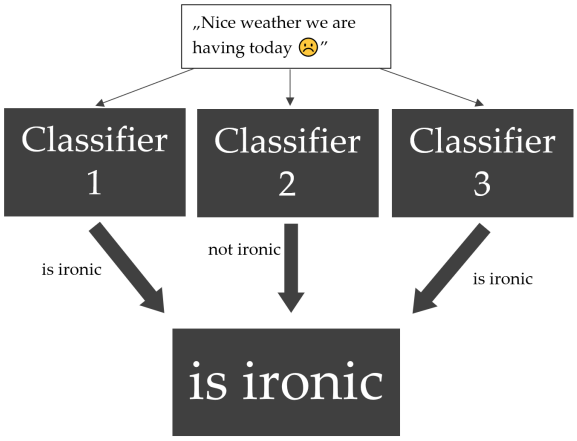

Figure 1: Ensemble approach to irony detection using simple majority voting.

3.3. Transformer Model

Related work has shown that classifiers for this task must be able to handle the language style and emojis from Twitter data properly. In addition to that, transformer-based approaches have been shown to outperform alternatives. Therefore, we consider BERTweet (Nguyen et al., 2020) to be the most suitable backend model for this task. It is a publicly available, large-scale language model and pre-trained for English Tweets using the RoBERTa pre-training procedure (Liu et al., 2019). This pre-training approach trains the model longer and utilizes other methods to be more optimized and robust than BERT-base (Devlin et al., 2019). BERTweet keeps BERT-base's model configuration. BERTweet-base was trained on 850M cased English tweets and outperforms RoBERTa and other competitors on downstream tweet NLP tasks.

3.4. Data augmentation

We propose three different ways of data augmentation in this work.[5]

Use of antonyms and negating. We argue that it is possible to generate new labeled training data by converting "ironic by polarity clash" cases to "not ironic". This happens by replacing words with strong explicit sentiment with their antonyms, i.e. a word with the opposite meaning (e.g. "love" $\rightarrow$ "hate"). This is likely to remove the polarity clash in an expression because two contrasting sentiment expressions are now either two positive or two negative expressions. This method is not able to change the polarity of an implicitly expressed sentiment because they usually lack words that have an explicit sentiment. It would still work for tweets that contain one implicit and one explicit sentiment expression. Tweets, where emojis are used, must first be converted to their textual counterpart because they are used to deliver sentiment as well (e.g. the word "joy" in "face with tears of joy"). To not overfit the classifier with too many similar sentences, this replacement method is only applied once for each tweet even if more possibilities for other antonyms exist. Which word is replaced with an antonym will be decided randomly.

For cases where the system is not able to find strong sentiment words or can not generate antonyms, negation seems to be a possible tool. The word "not" will be placed in front of verbs. This method might not be as reliable as the previous one because it can miss auxiliary verbs. Simple negation is also a weaker form of an antonym. The application of these methods shifts the distribution of training data further towards a non-ironic majority but it can help the transformer model to focus more on the presence of a polarity clash. It then might perform better in this category.

To the best of our knowledge, these augmentation methods have not been used yet in this context. They are only useful because the annotation allows it. It would not work on datasets with only a binary label because situational irony and other cases would be incorrectly modified.

We should also add that there are many more ways that augmentation can be approached in the context of irony detection. Our motivation to focus on the negative class emerged from the classification scheme, in particular the 'polarity clash' class. This leaves plenty of scope for future work exploring other augmentation methods.

Back-translation. The method of back translation of samples increases the amount of data by adding slightly modified copies. This approach has proven to be useful in Lee et al. (2020). Since there are no findings reported in the literature as to what languages might be best when doing back translation, the following eight European languages were chosen: Spanish, Finnish, Russian, Polish, German, Czech, Dutch and French.

These back-translated cases can be used for separate classifiers, the replacement with antonyms or to balance the class distributions in another classifiers' training data.

Tuning class-distribution. The back-translation also allows the implementation of a model with a more balanced class distribution. The ratio of training cases of non-ironic to, e.g., situational irony is at about 1:6. Adding different versions of the back-translated datasets and thus lowering the ratio could also improve the classification. Using oversampling and adjusting the class weights was found to improve the classification performance in this task (Van Hee et al., 2018).

[5]For implementational details please refer to the code on GitHub.

3.5. Putting it all together

Figure 2 depicts the idea of an ensemble incorporating the various forms of data augmentation.

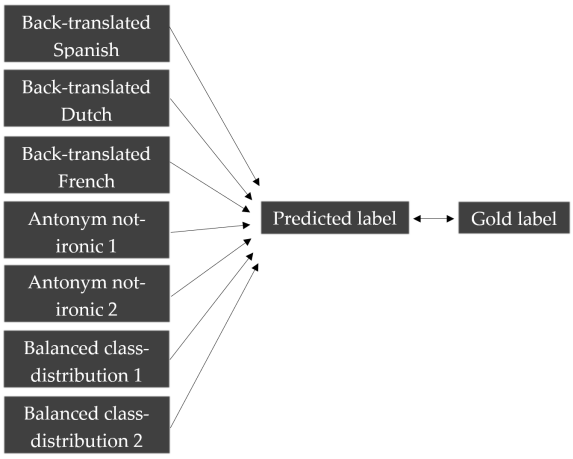

Figure 2: Ensemble approach to irony detection using different forms of data augmentation.

4. Experimental Setup

To measure the impact of the data augmentation methods, multiple models will be trained and compared to the *baseline of the unmodified training set*.

Per language, one model will be fine-tuned on the 3834 baseline training cases and an additional 3834 more from the back-translation process. Duplicate elements will be removed beforehand. Results will show if this addition increases or decreases the performance of a single model.

Other models will be trained with the baseline dataset and additional non-ironic expressions from the antonym and negation data generation method, or having a balanced class distribution. Randomization in the antonym data generation method yields different modified tweets in each process. Because of this, ten models will be trained for each case with the same augmentation method and parameter settings. Each fine-tuned model will then be put in an ensemble classifier with majority voting (as depicted in Figure 2) to measure how they perform. This will reveal whether they all give similar results for each test case or if there is variation between them.

The performance of these ensembles will be one of the decisive factors if a model will be used in the final ensemble. It will be a combination of the best-performing models. Additionally for the potentially best ensemble, a cross-validation will be performed to measure the model's robustness.

As discussed previously, BERTweet will be used for fine-tuning in all cases. Table 1 lists the hyperparameters used for this process.

Hyperparameter	Value
Batch size	16
Validation batch size	32
Epochs	4
Optimizer	Adam
Learning rate	2e-5
Epsilon	1e-8
Clipnorm	1.0
Epsilon	1e-8

Table 1: BERTweet Hyperparameter settings

5. Evaluation Methodology

To compare the different models, macro-F1 scores will be used in line with SemEval 2018 Task 3.

Shapiro-Wilk tests will be used to measure whether a set of scores is normally distributed, even if they have low statistical power on small sample sizes. Depending on the presence of a normal distribution, either a paired t-test or a Wilcoxon signed-rank test will be performed to compare two sets. If the test's p values are below an alpha of 0.05, the null hypothesis that two sets have the same distribution can be rejected. Tests for significance between our own and results of other papers reported in the literature are not possible because we did not have access to the implementations and have to compare against a single reported result.

6. Results

We will discuss the main results, tables with full results of all experimental settings can be found on the corresponding GitHub account.

6.1. Binary Classification

Table 2 summarises the main results for Task A (binary classification) with comparison to state-of-the-art performance as reported in the literature. We refer to the top three systems out of 43 runs submitted to the original challenge but also include more recent SOTA systems.[6]

For the binary irony classification task, the proposed methods appear to be marginally different from top results in the literature (substantially higher though than any of the official SemEval 2018 submissions). In particular, the simple ensemble appears to be on par with what Singh et al. (2019) and Potamias et al. (2020) were able to achieve. Interestingly, the results suggest that in this setup and context data augmentation did not seem to help in the case of binary classification.

The proposed model which uses different types of transformers in an ensemble is not better than ensembles which use the same type of model multiple times. The ensemble of the baseline models, the ensemble

[6]Results for Baziotis et al. (2018) for both Task A and Task B are a lot higher than their official SemEval runs and we report their post-task-completion work.

of models with additional back-translated data and the combination approach of language, antonyms, and balanced datasets show very similar results.[7]

System	F1 (Task A)
Wu et al. (2018)	0.705
Baziotis et al. (2018)	0.786
Rohanian et al. (2018)	0.650
Nguyen et al. (2020)	0.746
Singh et al. (2019)	**0.803**
Potamias et al. (2020)	0.80
Ensemble baseline	0.798
Ensemble language	0.794
Ensemble combination	0.784

Table 2: F1 Scores for Task A of related work and own models (best marked up in bold)

6.2. Multiclass Classification

Table 3 reports results for Task B (multi-label classification), again comparing against state-of-the-art performance reported in the literature. Again we first report the three top-performing systems (out of 31) in the original submission as well as a more recent state-of-the-art system.

Here we find that our proposed methodology outperforms existing alternatives for the task of multiclass irony detection by a large margin. Compared to (Ghosh and Veale, 2018) the increase in performance is particularly noticeable in the cases of classification of irony by polarity clash and situational irony.

The contribution of data augmentation is inconclusive overall but for the aggregated F1 score we observe that back-translation did push up the performance. This leaves scope for future work on other datasets.

7. Discussion

We found that the use of data augmentation methods allows single transformer models to achieve results with a lower standard deviation at the cost of performance.

We also conclude for the experiments we conducted that the method of negating the ironic statement by sentiment clash tweets has not shown to be successful. This might be because the assumption of removing the ironic nature by negation does not generally hold true. After all, it can not generate the correct polar opposite all the time. A more sophisticated approach might be needed here. Also, further qualitative analysis of the automatically augmented data might provide more insights into this finding.

The approach with the replacement of words with antonyms was shown to be more successful.

[7]Detailed result tables and significance tests can be found on the project repository. They give more detailed insights into what classifiers work how well overall and in which class helping to understand the contribution of each augmentation method.

Generally though, our results would support the finding that the use of data augmentation is not necessarily needed because the ensemble system which uses the original training data achieves similar results with no statistical difference.

Possible improvements for approaches in this work can be the search for better hyperparameters for the fine-tuning process. E.g., a grid search can be performed to find the most optimal batch size, training epochs etc. It can also be tested if a weighted ensemble classifier has superior performance over the current majority voting system. There is also a need for a method that can generate more ironic cases when only a small dataset of ironic data is available. This can, for example, be a GPT model that can produce more tweet-like sentences which contain one of the infrequent irony cases like situational irony (Floridi and Chiriatti, 2020). A program that takes one part of a sentence and adds an unexpected outcome might also be possible for this scenario.

We can now revisit our research questions and answer them as follows.

- **RQ 1**: *Is an ensemble classifier combining fine-tuned transformer models able to achieve better results in irony detection than state-of-the-art baselines?*

  We find that in the case of fine-grained irony detection ensembles of fine-tuned transformer models can beat existing state-of-the-art approaches (for the chosen reference dataset) This was not found for binary irony classification though.

- **RQ 2**: *Does the use of data augmentation methods improve classification quality?*

  Data augmentation has helped establish state-of-the-art performance for fine-grained classification but has failed to demonstrate that for the binary case.

We should point out one limitation of this work and that is the use of just one reference dataset. However, there are two reasons for that. First of all, it is considered to be the main benchmark collection for the detection of irony in English. It remains to be seen what future work shows in regards to applying the techniques to new datasets but also how the outlined data augmentation ideas transfer to different languages. Secondly, we did point out problems with getting hold of datasets that go beyond the simply binary distinction of ironic *vs.* non-ironic. Clearly more effort is needed to push forward the state of the art in this field. We see the provision of all code, results and statistical analysis on our project repository as a contribution that helps with reproducibility but also lowers the entry barrier into developing other solutions to the problem by building on the code provided.

System	F1 (Task B)	Not ironic	Irony by clash	Situational irony	Other
Baziotis et al. (2018)	0.5358	-	-	-	-
Ghosh and Veale (2018)	0.507	0.843	0.697	0.376	0.114
Wu et al. (2018)	0.495	0.704	0.608	0.433	0.233
Singh et al. (2019)	0.5565	-	-	-	-
Ensemble baseline	0.590	**0.847**	**0.782**	0.591	0.141
Ensemble language	**0.611**	0.835	0.764	0.599	**0.244**
Ensemble combination	0.600	0.820	0.764	**0.617**	0.190

Table 3: F1 scores for Task B of related work and own models (per column best marked up in bold)

8. Conclusions

We have looked at automatic irony detection in social media, a challenging NLP problem which is still hard to crack. We explored ensemble methods and data augmentation to address both the binary as well as the fine-grained irony detection problem. We conclude that both of these ideas are very promising directions for future work.

There are however broader issues that need to be explored. For example, we argue that standard machine learning metrics are problematic in the context of irony detection. Humans do not have a high inter-agreement score when annotating irony, another evaluation metric might be more suitable to allow a better comparison with the computer's score compared to a human's score in classification.

As motivated before, annotation schemes should classify into multiple irony and sarcasm types, e.g. "insulting" and "not insulting" sarcasm (though one might argue that sarcasm is insulting per se). However, any more fine-grained annotation scheme is likely to make the task even harder and lower the inter-annotation agreement scores even further. In general, we conclude that multiclass irony detection looks to be too challenging still to be applied in real-world applications.

9. Ethical Considerations

Processing personal data such as social media posts always raises concerns about ethical issues. We do however not see any reason for concern as we are using a common benchmark collection which uses data that had already been de-personalised.

Acknowledgements

This work was supported by the project *COURAGE: A Social Media Companion Safeguarding and Educating Students* funded by the Volkswagen Foundation, grant number 95564. We would also like to thank the three anonymous reviewers for detailed and constructive feedback.

10. Bibliographical References

Aggarwal, C. C. and Zhai, C. (2012). *Mining text data*. Springer Science & Business Media.

Attardo, S. (2007). Irony as relevant inappropriateness. *Irony in language and thought*, pages 135–174.

Barbe, K. (1995). *Irony in context*, volume 34. John Benjamins Publishing.

Basile, V., Novielli, N., Croce, D., Barbieri, F., Nissim, M., and Patti, V. (2021). Sentiment Polarity Classification at EVALITA: Lessons Learned and Open Challenges. *IEEE Trans. Affect. Comput.*, 12(2):466–478.

Baziotis, C., Nikolaos, A., Papalampidi, P., Kolovou, A., Paraskevopoulos, G., Ellinas, N., and Potamianos, A. (2018). NTUA-SLP at SemEval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive RNNs. pages 613–621, June.

Bowes, A. and Katz, A. (2011). When sarcasm stings. *Discourse Processes*, 48(4):215–236.

Bueno, R. O., Pardo, F. M. R., Farías, D. I. H., Rosso, P., Montes-y-Gómez, M., and Medina-Pagola, J. (2019). Overview of the Task on Irony Detection in Spanish Variants. In *IberLEF@SEPLN*, volume 2421 of *CEUR Workshop Proceedings*, pages 229–256. CEUR-WS.org.

Carvalho, P., Sarmento, L., Silva, M. J., and De Oliveira, E. (2009). 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*, pages 53–56.

Cervone, A., Stepanov, E. A., Celli, F., and Riccardi, G. (2017). Irony detection: from the twittersphere to the news space. In *CLiC-it 2017-Italian Conference on Computational Linguistics*, volume 2006.

Cignarella, A. T., Frenda, S., Basile, V., Bosco, C., Patti, V., Rosso, P., et al. (2018). Overview of the EVALITA 2018 task on Irony Detection in Italian tweets (IronITA). In *Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2018)*, volume 2263, pages 1–6. CEUR-WS.

Clark, H. H. and Gerrig, R. J. (1984). *On the pretense theory of irony.* American Psychological Association.

Colebrook, C. (2002). *Irony in the Work of Philoso-*

phy. U of Nebraska Press.

Corrêa, U. B., Coelho, L., Santos, L., and de Freitas, L. A. (2021). Overview of the IDPT task on irony detection in portuguese at iberlef 2021. *Proces. del Leng. Natural*, 67:269–276.

Croes, E. A. and Antheunis, M. L. (2021). Can we be friends with Mitsuku? A longitudinal study on the process of relationship formation between humans and a social chatbot. *Journal of Social and Personal Relationships*, 38(1):279–300.

Dekker, K. and van der Goot, R. (2020). Synthetic data for English lexical normalization: How close can we get to manually annotated data? In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6300–6309, Marseille, France, May. European Language Resources Association.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Farías, D. I. H., Patti, V., and Rosso, P. (2016). Irony detection in Twitter: The role of affective content. *ACM Transactions on Internet Technology (TOIT)*, 16(3):1–24.

Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89.

Floridi, L. and Chiriatti, M. (2020). Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30(4):681–694.

Ghosh, A. and Veale, T. (2018). Ironymagnet at semeval-2018 task 3: A siamese network for irony detection in social media. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 570–575.

Ghosh, D., Vajpayee, A., and Muresan, S. (2020). A report on the 2020 sarcasm detection shared task. *arXiv preprint arXiv:2005.05814*.

González, J.-Á., Hurtado, L.-F., and Pla, F. (2019). Elirf-upv at irosva: Transformer encoders for spanish irony detection. In *IberLEF@ SEPLN*, pages 278–284.

González-Ibánez, R., Muresan, S., and Wacholder, N. (2011). Identifying sarcasm in Twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586.

Hagen, M., Potthast, M., Büchner, M., and Stein, B. (2015). Webis: An ensemble for Twitter sentiment detection. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 582–589.

Juuti, M., Gröndahl, T., Flanagan, A., and Asokan, N. (2020). A little goes a long way: Improving toxic language classification despite data scarcity. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2991–3009, Online, November. Association for Computational Linguistics.

Karoui, J., Benamara, F., Moriceau, V., Patti, V., Bosco, C., and Aussenac-Gilles, N. (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: Volume 1, Long Papers*. Association for Computational Linguistics (ACL).

Kreuz, R. and Caucci, G. (2007). Lexical influences on the perception of sarcasm. In *Proceedings of the Workshop on computational approaches to Figurative Language*, pages 1–4.

Lee, H., Yu, Y., and Kim, G. (2020). Augmenting data for sarcasm detection with unlabeled conversation context. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 12–17, Online, July. Association for Computational Linguistics.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Liu, B. (2020). *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge university press.

Maynard, D. and Bontcheva, K. (2015). Understanding climate change tweets: an open source toolkit for social media analysis. In *EnviroInfo/ICT4S (1)*, pages 242–250.

Maynard, D. G. and Greenwood, M. A. (2014). Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *LREC 2014 proceedings*. ELRA.

Mladenović, M., Ošmjanski, V., and Stanković, S. V. (2022). Cyber-aggression, cyberbullying, and cybergrooming: A survey and research challenges. *ACM Computing Surveys*, 54(1).

Nguyen, D. Q., Vu, T., and Nguyen, A. T. (2020). BERTweet: A pre-trained language model for English Tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Potamias, R. A., Siolas, G., and Stafylopatis, A.-G. (2020). A transformer-based approach to irony and

sarcasm detection. *Neural Computing and Applications*, 32(23):17309–17320.

Reyes, A., Rosso, P., and Buscaldi, D. (2012). From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering*, 74:1–12.

Rohanian, O., Taslimipoor, S., Evans, R., and Mitkov, R. (2018). Wlv at semeval-2018 task 3: Dissecting tweets in search of irony. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 553–559.

Rosenthal, S., Ritter, A., Nakov, P., and Stoyanov, V. (2014). SemEval-2014 Task 9: Sentiment Analysis in Twitter. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 73–80, 01.

Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S. M., Ritter, A., and Stoyanov, V. (2015). SemEval-2015 Task 10: Sentiment Analysis in Twitter. 451-463. In *Proc. of the 9th International Workshop on Semantic Evaluation*.

Sanguinetti, M., Poletto, F., Bosco, C., Patti, V., and Stranisci, M. (2018). An Italian Twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Singh, A., Blanco, E., and Jin, W. (2019). Incorporating emoji descriptions improves tweet classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2096–2101.

Sulis, E., Irazú Hernández Farías, D., Rosso, P., Patti, V., and Ruffo, G. (2016). Figurative messages and affect in twitter. *Know.-Based Syst.*, 108(C):132–143, sep.

Tran, H. N. and Kruschwitz, U. (2021). ur-iw-hnt at GermEval 2021: An ensembling strategy with multiple BERT models. In *Proceedings of the GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments*, pages 83–87, Duesseldorf, Germany, September. Association for Computational Linguistics.

Utsumi, A. (1996). A unified theory of irony and its computational formalization. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Van Hee, C., Lefever, E., and Hoste, V. (2016). Guidelines for annotating irony in social media text, version 2.0. *LT3 Technical Report Series*.

Van Hee, C., Lefever, E., and Hoste, V. (2018). Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50.

Van Hee, C. (2017). *Can machines sense irony?: exploring automatic irony detection on social media*. Ph.D. thesis, Ghent University.

Wang, P.-Y. A. (2013). #irony or #sarcasm — a quantitative and qualitative study based on Twitter. In *Proceedings of the 27th Pacific Asia Conference on Language, Information, and Computation (PACLIC 27)*, pages 349–356, Taipei, Taiwan, November. Department of English, National Chengchi University.

Whitfield, D. (2021). Using GPT-2 to create synthetic data to improve the prediction performance of NLP machine learning classification models. *CoRR*, abs/2104.10658.

Wilson, D. and Sperber, D. (1992). On verbal irony. *Lingua*, 87(1):53–76.

Wu, C., Wu, F., Wu, S., Liu, J., Yuan, Z., and Huang, Y. (2018). THU_NGN at SemEval-2018 task 3: Tweet irony detection with densely connected LSTM and multi-task learning. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, New Orleans, Louisiana, June. Association for Computational Linguistics.

Wullach, T., Adler, A., and Minkov, E. (2021). Fight fire with fire: Fine-tuning hate detectors using large samples of generated hate speech. In Marie-Francine Moens, et al., editors, *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 4699–4705. Association for Computational Linguistics.

Zimmerman, S., Kruschwitz, U., and Fox, C. (2018). Improving hate speech detection with deep learning ensembles. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.