Fine Irony: Training Transformers with Ordinal Likelihood Labels

Anonymous ACL submission

Abstract

In this paper, we investigate a selection of methodologies for fine-tuning transformer-based classifiers for fine-grained irony (likelihood) detection in English Tweets. These methodologies include approaching irony detection as an ordinal classification task and as a regression task for varying label granularity (3, 5 and 7 labels). Our experiments show that training irony detection models using fine-grained likelihood labels is not only possible but also advantageous, as the models reach higher F1-scores for binary classification than models that are specifically trained for this task. In addition, we explore how well the predictions by each of the model setups can be interpreted through Layer Integrated Gradients. The results show that, although performance for irony detection is consistent, the selection of important words for well-performing models does not consistently align with human trigger word annotation.

1 Motivation & Related Work

The detection of irony and sarcasm relies on identifying whether an utterance should be interpreted literally or if the opposite meaning should be assumed. Traditionally (Filatova, 2012; Van Hee et al., 2018; Oprea and Magdy, 2020) and recently still (Misra and Arora, 2023; Rahma et al., 2023), this task is approached as a binary classification problem. To some extent, that is a sensible decision, since this type of figurative language inverts the literal meaning of an utterance and something is either ironic or not. However, humans need to rely on extra-textual background information to evaluate whether a statement should be interpreted literally. While such extra-textual knowledge is mostly shared, it never completely overlaps as it can include (specific) objective knowledge as well as subjective opinions and assumptions. As a result, humans have to rely on the assumption of shared knowledge with some degree of confidence. This confidence about shared knowledge directly transfers to the confidence of the irony prediction. For this reason, confidence has been included in a few annotation schemes for irony detection (Van Hee et al., 2018; Wallace et al., 2014). Although they suggest weighting irony estimation with the confidence about the shared background knowledge (Wallace, 2015), the merits of this suggestion have not been fully explored. To our knowledge, the only related development in recent work is the annotation of irony likelihood labels, which merge confidence with binary classification labels (Maladry et al., 2024). In this novel scheme, the 7 labels describe samples as "Definitely Not Ironic", "Probably Not Ironic", "Rather Not Ironic", "Not sure", "Rather Ironic", "Probably Ironic" and "Definitely Ironic". The authors also showed that the fine-grained annotation scheme notably does not harm inter-rater agreement and provides more nuanced labels. As a result, the incorporation of these nuanced labels becomes a promising research avenue for irony detection.

2 Methodology

We evaluate our approaches by fine-tuning two pre-trained models with the original data distribution of 3451 train, 383 development and 958 test samples from the irony data set released by Van Hee et al. (2018). The first model, BERTweet (Nguyen et al., 2020) is trained on social media data. The second model, DeBERTa v3 (He et al., 2023), is more recent and reaches promising scores for a wide variety of tasks, but it is not necessarily adapted for social media data. Both models have shown to perform well for irony detection (Farha et al., 2022).

For the baselines, we fine-tuned the pre-trained models for binary irony classification. This means merging variants of "*Not Ironic*" labels and the

Num.	7	5	3	2
7	Iron.	Iron.	Iron.	Iron.
6	Prob. Iron.	Iron.	Iron.	Iron.
5	Rath. Iron.	Rath. Iron.	Iron.	Iron.
4	Not Sure	Not Sure	Not Sure	˜Iron.
3	Rath.˜Iron.	Rath.˜Iron.	˜Iron.	˜Iron.
2	Prob.˜Iron.	˜Iron.	˜Iron.	˜Iron.
1	˜Iron.	˜Iron.	˜Iron.	˜Iron.

Table 1: Original label numbers and their corresponding classes after merging to coarse-grained granularity. "*Not Ironic*" labels are indicated as ˜Ironic.

"*Not Sure*" label into a single negative class. All other labels are then considered ironic. Each model is fine-tuned for up to 10 epochs with a learning rate of 5e-6, weight decay of 0.01 and a batch size of 16. Training parameters (including 200 warm-up steps) are updated every 100 steps. Early stopping is implemented based on development loss to prevent overfitting. To ensure that these results can be generalized and reproduced, we maintain the same parameters for all models and start from 6 specific seeds, for which the results are averaged.[1] As the goal of this paper is to make use of the fine-grained annotations during training, we propose redefining irony detection as two new tasks: ordinal classification and regression. For each task, we explore the different strategies to merge the 7 labels into 5, 3 or 2 coarse-grained labels (see Table 1).

2.1 Ordinal Classification

For ordinal classification, we use each of the labels as a separate class. Since the fine-grained labels are ordered, the distance between two labels should be taken into account. The distance between "1: Definitely Not Ironic" and "6: Probably Ironic", for example, should be greater than the distance between "3: Rather Not Ironic" and "5: Rather Ironic". To incorporate this into the training procedure, we make use of a weight matrix and calculate the log ordinal loss (Equation 1), which has shown to be effective for ordinal classification (Castagnos et al., 2022).

$$\mathcal{L}_{OLL-\alpha}(P, y) = \sum \log(1 - p_i) d(y, i)^{\alpha} \quad (1)$$

[1]The best-performing models and the corresponding code will be made publicly available on HuggingFace and GitHub.

2.2 Regression

One of the major downsides of this ordinal classification model, is that it considers the labels as separate classes rather than the continuum that it is supposed to reflect. Therefore, we also propose the use of a regression model, which treats the task as single continuous value instead of log probabilities for each individual label. Whereas mean squared error loss is the default implemented for transformer models, exploratory testing revealed that employing Huber loss (Equation 2, with δ as a hyper-parameter for Hinge loss), tends to result in better scores.

$$\mathcal{L}_{\delta} = \begin{cases} \frac{1}{2}(y - \hat{y})^2 & if\ |(y - \hat{y})| < \delta \\ \delta((y - \hat{y}) - \frac{1}{2}\delta) & otherwise \end{cases} \quad (2)$$

3 Results

3.1 Fine-grained Labels

As shown in Table 2, models fine-tuned for regression consistently obtain a lower mean squared error (MSE) for both pre-trained models across all 3, 5 and 7-label granularity compared to the ordinal classification models. As the regression models are not restricted to predicting the exact classes, the free-range predictions allow them to attain more accurate scores for predicting the fine-grained likelihood labels.

Investigation of the different pre-trained base models does not reveal that any model is consistently superior to the other. Still, there are some slight differences. For regression models, BERTweet has lower MSE scores for 3-label and 7-label granularity, but higher loss for the 5-label granularity. For ordinal classification models, DeBERTa has lower loss for 5 and 7-label granularity but higher loss for 3-label granularity.

3.2 Binary Classification

Whereas the evaluation through MSE indicates how well our models are able to project their predictions on the irony + confidence spectrum, it remains important to also evaluate for binary classification, where the boundary between "ironic" or "not ironic" is more absolute. The F1-scores in Table 3 reveal that the regression models (and binary classification models) function better with BERTweet, whilst DeBERTa reaches higher scores for ordinal classification models. Furthermore, both regression

Task	Gran.	BaseModel	MSE
regression	3	BERTweet	0.676
regression	3	DeBERTa	0.681
ordinal	3	BERTweet	0.787
ordinal	3	DeBERTa	0.787
regression	5	BERTweet	2.721
regression	5	DeBERTa	2.691
ordinal	5	BERTweet	3.115
ordinal	5	DeBERTa	2.846
regression	7	BERTweet	5.308
regression	7	DeBERTa	5.385
ordinal	7	BERTweet	6.138
ordinal	7	DeBERTa	5.477

Table 2: MSE scores for ordinal and regression for varying label granularity.

Task	Gran.	BaseModel	F1
regression	5	BERTweet	0.765
regression	7	BERTweet	0.764
ordinal	5	DeBERTa	0.758
ordinal	7	DeBERTa	0.756
ordinal	5	BERTweet	0.755
binary	2	BERTweet	0.755
ordinal	7	BERTweet	0.754
regression	3	BERTweet	0.752
regression	5	DeBERTa	0.749
binary	2	DeBERTa	0.743
ordinal	3	BERTweet	0.741
regression	7	DeBERTa	0.734
regression	3	DeBERTa	0.725
ordinal	3	DeBERTa	0.549

Table 3: F1-score ranking for all models averaged across random seeds. Baseline models are indicated in gray.

and ordinal models outperform binary classification models in their optimal setup (best base model) and either a 5-point or 7-point granularity. Regarding the label granularity, we conclude that the 5-point granularity is generally more performative than the full 7-point granularity.

Closer evaluation of the scores, shown in Figure 1a, reveals that all models perform better on the "not ironic" labels (1, 2, 3, 4) than on the "ironic" labels (5, 6, 7). Likely, this is connected to the distribution of the dataset, which has 64% not-ironic samples and 36% ironic samples. Whilst all models exhibit this performance difference, the discrepancy is most prevalent for the regression models, smaller for ordinal models and the smallest for the binary classification model. Within the non-ironic and ironic categories, the scores are the best for the highest-confidence labels 1 and 7, while they are lower on the in-between labels (2, 3) and (5, 6), as illustrated in Figure 1b.

3.3 Interpretability

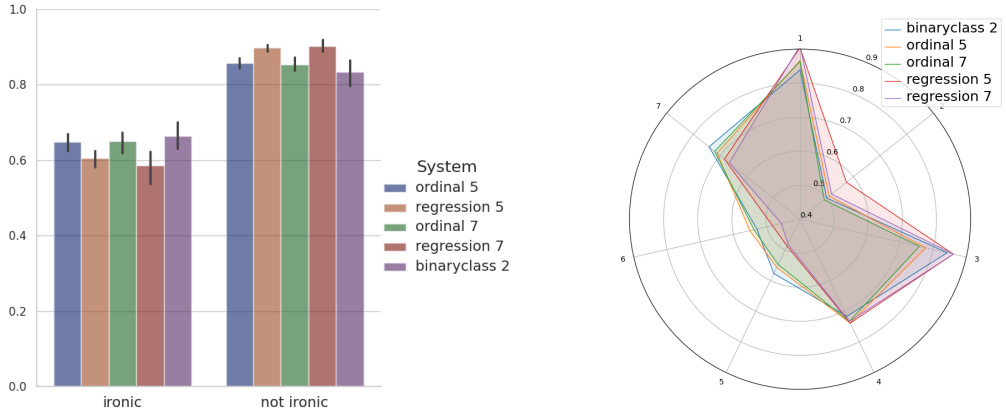
Finally, we investigate whether the more nuanced labels make the reasoning of fine-tuned models more human-like. To approximate system reasoning, we employ Layer Integrated Gradients (Sundararajan et al., 2017) to generate post-hoc numerical importances for all sub-tokens, which are then mapped to the word level based on space splitting. To evaluate how closely these metrics resemble human reasoning, we make use of Accumulated Precise Importance (Maladry et al., 2024). This is a sentence-level metric that sums up the (normalized) numerical importances for each trigger word token (based on human annotations). To illustrate, the API score for Example 1 equals 95% (33 + 18 + 19 + 25), with only 5% of the total importances being attributed to a non-trigger word.

Example 1

	love	*getting*	*my*	*papers*	*rejected*	*:')*
HUM.	*1*	*0*	*0*	*1*	*1*	*1*
IMP.	*.33*	*0*	*.05*	*.18*	*.19*	*.25*

As the explanations for irony only matter for ironic tweets (non-ironic tweets do not contain specific markers), we only calculate the API scores for correctly predicted ironic tweets and average them.

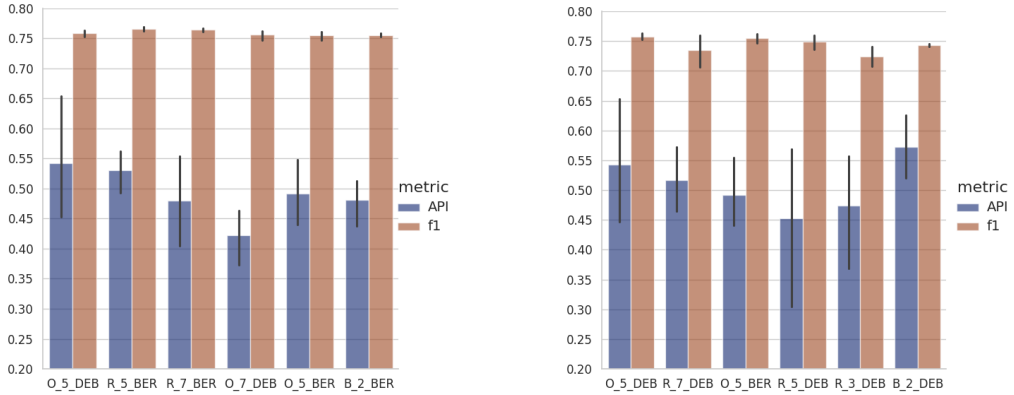
As shown in Figure 2a, the models that attained the highest F1-scores for binary classification do not consistently achieve the highest API scores. Although some models, such as the Ordinal DeBERTa model with 5-label granularity (O_5_DEB), perform well on API-scores and F1-score, the high standard deviation (black line) indicates that the interpretability of the specific model is highly variable. In this case, the only difference between the same-type models is the random seed they are trained with (detailed results in Table 4 Appendix A). However, there is a large discrepancy with the lowest score achieved by the same model trained with a different random seed (33.4%). Furthermore, the best average API scores are attained by a binary classification model trained with DeBERTa (see Figure 2b).



(a) Ironic vs Not Ironic

(b) Complete 7-label granularity

Figure 1: Accuracy scores on specific labels (averaged across both models and all random seeds).



(a) Best models based on F1 scores

(b) Best models based on API scores

Figure 2: F1 and API scores for best-performing fine-grained + binary classification models (see top of Table 3).

3.4 Discussion

Our experiments indicate that our models trained using likelihood labels do not only provide more nuanced outputs, but once those outputs are merged to binary labels, they can also perform better at binary irony classification compared to models that are trained specifically for binary classification. The additional information encoded in the labels, in particular for the 5-label and 7-label granularity, proves to be valuable during training.

Whilst F1-scores achieved by the models seems to be consistent across random seeds, the API-scores vary significantly. This means that the very same model can develop different "reasoning" patterns depending on random initialization. This may be important to keep in mind for applications where the systems should be human-interpretable and exhibit intuitive trigger word patterns. Still, even if a model has developed different reasoning patterns, this does not necessarily mean that the patterns are invalid, if they help the model to arrive at the same conclusion.

4 Conclusion

With this research, we have successfully trained systems for irony likelihood prediction and irony detection based on a novel irony likelihood dataset. Moreover, we found that regression models trained with 5 and 7-label granularity even outperform binary classifiers on binary irony detection. Our interpretability experiments demonstrate that model performance and interpretability do not always align. Using current approaches, training a model for (irony) prediction without explanation does not guarantee reliable interpretability. For this reason, we suggest incorporating interpretability in the fine-tuning process for classification tasks in future research.

Limitations

In this paper, we investigated how fine-grained labels for irony detection can be leveraged during transformer fine-tuning. The primary limitation of our research lies in the dataset itself, which relies on the annotations of a single individual. As shown by related work, the perception of irony can be highly subjective and can also be dependent on the linguistic and communicative skills of the writer of a text. This means that the labels for the trigger words and irony labels are open to interpretation and should not be considered 100% correct or incorrect. A secondary limitation of our research is that we do not investigate the latest large generative LMs, which have become the state-of-the-art for many tasks.

References

François Castagnos, Martin Mihelich, and Charles Dognin. 2022. A simple log-based loss function for ordinal text classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4604–4609, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Ibrahim Abu Farha, Silviu Oprea, Steve Wilson, and Walid Magdy. 2022. Semeval-2022 task 6: isarcasmeval, intended sarcasm detection in english and arabic. In *The 16th International Workshop on Semantic Evaluation 2022*, pages 802–814. Association for Computational Linguistics.

Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Lrec*, pages 392–398. Citeseer.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Aaron Maladry, Alessandra Teresa Cignarella, Els Lefever, Cynthia van Hee, and Veronique Hoste. 2024. Human and system perspectives on the expression of irony: An analysis of likelihood labels and rationales. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8372–8382, Torino, Italia. ELRA and ICCL.

Rishabh Misra and Prahal Arora. 2023. Sarcasm detection using news headlines dataset. *AI Open*, 4:13–18.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 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*, pages 9–14.

Silviu Oprea and Walid Magdy. 2020. iSarcasm: A dataset of intended sarcasm. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1279–1289, Online. Association for Computational Linguistics.

Alaa Rahma, Shahira Shaaban Azab, and Ammar Mohammed. 2023. A comprehensive review on arabic sarcasm detection: Approaches, challenges and future trends. *IEEE Access*.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR.

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

Byron C Wallace. 2015. Computational irony: A survey and new perspectives. *Artificial intelligence review*, 43:467–483.

Byron C Wallace, Laura Kertz, Eugene Charniak, et al. 2014. Humans require context to infer ironic intent (so computers probably do, too). In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 512–516.

Appendix

A Detailed Results

Table 4: Results for each individual regression, ordinal and binary classification model for different random seeds and label granularity.

Task	Gran.	BaseModel	Seed	MSE	F1	API
ord.	5	DeBERTa	666	2.82	0.76	0.78
bin.	2	DeBERTa	666		0.74	0.67
regr.	5	DeBERTa	666	2.80	0.75	0.66
bin.	2	DeBERTa	69		0.74	0.65
ord.	5	BERTweet	666	3.30	0.74	0.63
ord.	5	DeBERTa	13	2.76	0.77	0.62
regr.	7	DeBERTa	42	5.44	0.71	0.61
regr.	3	DeBERTa	42	0.69	0.71	0.60
regr.	7	DeBERTa	666	5.51	0.75	0.60
regr.	7	BERTweet	69	5.20	0.77	0.59
regr.	3	BERTweet	7	0.69	0.74	0.59
regr.	7	BERTweet	13	5.14	0.76	0.58
ord.	3	BERTweet	666	0.79	0.74	0.58
ord.	7	BERTweet	420	5.90	0.76	0.58
regr.	5	BERTweet	666	2.68	0.77	0.58
regr.	5	BERTweet	7	2.78	0.76	0.57
bin.	2	DeBERTa	13		0.75	0.57
regr.	5	DeBERTa	7	2.73	0.75	0.57
regr.	5	BERTweet	420	2.78	0.77	0.57
regr.	3	BERTweet	420	0.71	0.75	0.56
ord.	3	BERTweet	420	0.79	0.74	0.56
regr.	3	DeBERTa	420	0.67	0.74	0.55
bin.	2	DeBERTa	42		0.75	0.55
ord.	3	DeBERTa	7	0.77	0.69	0.55
ord.	3	BERTweet	13	0.76	0.74	0.54
ord.	5	DeBERTa	420	2.86	0.77	0.54
ord.	3	BERTweet	42	0.80	0.73	0.53
regr.	3	BERTweet	42	0.60	0.72	0.53
bin.	2	BERTweet	666		0.75	0.53
ord.	5	BERTweet	69	3.00	0.76	0.53
bin.	2	DeBERTa	7		0.74	0.53
ord.	3	DeBERTa	13	0.80	0.74	0.53
ord.	7	BERTweet	666	6.08	0.75	0.53
regr.	3	DeBERTa	666	0.66	0.75	0.52
bin.	2	BERTweet	42		0.76	0.52
ord.	7	BERTweet	13	6.13	0.76	0.52
regr.	3	DeBERTa	7	0.73	0.75	0.51
regr.	5	BERTweet	69	2.74	0.76	0.51
regr.	7	DeBERTa	7	5.29	0.73	0.51
ord.	7	BERTweet	42	6.16	0.76	0.51

regr.	5	BERTweet	42	2.62	0.77	0.51
regr.	3	BERTweet	666	0.68	0.76	0.50
regr.	5	DeBERTa	13	2.67	0.75	0.50
bin.	2	BERTweet	7		0.76	0.50
ord.	5	DeBERTa	69	2.82	0.76	0.50
ord.	3	BERTweet	69	0.78	0.75	0.50
regr.	3	BERTweet	13	0.69	0.76	0.50
regr.	7	BERTweet	666	5.10	0.77	0.49
regr.	7	BERTweet	7	5.41	0.77	0.49
ord.	5	BERTweet	420	3.06	0.76	0.49
ord.	7	DeBERTa	666	5.33	0.76	0.49
bin.	2	BERTweet	420		0.76	0.48
regr.	7	DeBERTa	13	5.30	0.77	0.48
ord.	7	DeBERTa	7	5.62	0.75	0.48
bin.	2	BERTweet	13		0.75	0.48
ord.	7	BERTweet	69	6.33	0.74	0.48
bin.	2	DeBERTa	420		0.74	0.47
ord.	5	BERTweet	42	3.20	0.75	0.46
regr.	5	BERTweet	13	2.72	0.76	0.46
regr.	5	DeBERTa	420	2.67	0.76	0.46
regr.	7	DeBERTa	69	5.58	0.68	0.45
regr.	3	BERTweet	69	0.69	0.77	0.45
regr.	7	DeBERTa	420	5.18	0.77	0.45
ord.	7	DeBERTa	420	5.43	0.75	0.44
ord.	7	DeBERTa	13	5.55	0.76	0.44
regr.	3	DeBERTa	13	0.66	0.71	0.44
ord.	7	BERTweet	7	6.24	0.75	0.44
ord.	5	BERTweet	13	2.95	0.76	0.43
ord.	5	DeBERTa	42	2.83	0.75	0.43
ord.	3	DeBERTa	420	0.78	0.73	0.42
ord.	3	BERTweet	7	0.80	0.74	0.41
ord.	5	BERTweet	7	3.19	0.75	0.41
ord.	5	DeBERTa	7	2.98	0.75	0.39
regr.	5	DeBERTa	42	2.75	0.72	0.39
bin.	2	BERTweet	69		0.75	0.38
regr.	7	BERTweet	42	5.33	0.77	0.37
regr.	7	BERTweet	420	5.66	0.76	0.36
ord.	7	DeBERTa	69	5.26	0.77	0.36
ord.	7	DeBERTa	42	5.68	0.74	0.33
regr.	3	DeBERTa	69	0.68	0.69	0.22
regr.	5	DeBERTa	69	2.52	0.76	0.14
ord.	3	DeBERTa	42	0.77	0.38	0.00
ord.	3	DeBERTa	69	0.80	0.38	0.00
ord.	3	DeBERTa	666	0.80	0.38	0.00

Acknowledgments

We would like to thank the reviewers in advance for their constructive feedback and suggestions for improving our work.