A Fine Line Between Irony and Sincerity: Identifying Bias in Transformer Models for Irony Detection

Aaron Maladry, Els Lefever, Cynthia Van Hee and **Veronique Hoste**
LT3 - Ghent University
firstname.lastname@ugent.be

Abstract

In this paper we investigate potential bias in fine-tuned transformer models for irony detection. Bias is defined in this research as spurious associations between word n-grams and class labels that can cause the system to rely too much on superficial cues and miss the essence of the irony. For this purpose, we looked for correlations between class labels and words that are prone to trigger irony, such as positive adjectives, intensifiers and topical nouns. Additionally, we investigate our irony model's predictions before and after manipulating the data set through irony trigger replacements. We further support these insights with state-of-the-art explainability techniques (Layer Integrated Gradients, Discretized Integrated Gradients and Layer-wise Relevance Propagation). Both approaches confirm the hypothesis that transformer models generally encode correlations between positive sentiments and ironic texts, with even higher correlations between vividly expressed sentiment and irony. Based on these insights, we implemented a number of modification strategies to enhance the robustness of our irony classifier.

1 Introduction

Irony is a complex form of figurative language with which people convey the opposite meaning of what they say. A typical example of verbal irony is the explicit expression of positive sentiment towards a negative situation or event. In some ironic statements, both the positive sentiment and the negative sentiment are expressed explicitly, like in the following example: "So nice of my stupid neighbor to start mowing the lawn in the morning". However, more subtle ironic statements make this paradox less obvious, if the speaker leaves out the explicit negative sentiment ("stupid"). When doing so, they assume the receiver of their message already knows the connotative sentiment linked to this situation. This assumption of connotative common-sense knowledge, along with the contradicting nature of the expression, makes automatic irony detection a notoriously hard task. Since detecting irony can also be difficult for humans, we often use rhetorical devices such as exaggerations, metaphors and intonation (in spoken language) or tone to hint at the underlying irony.

Whereas traditional feature-based approaches have long been the go-to methodology, most state-of-the-art systems have switched to bi-directional transformers (Devlin et al., 2018). These transformer systems have taken the lead for most benchmarks in Natural Language Processing (NLP), thanks to their word and sentence representations. This leap did, however, come at the cost of model explainability and insights into feature importance, which are more easily accessible for traditional machine learning algorithms, such as Logistic Regression and Decision Tree.

In this paper, we explore several explainability techniques and identify patterns in computational modeling of a fine-tuned transformer model for irony detection. In addition to existing metrics, we first search for potential biases in the data based on correlations between the irony label and lemmas in the train data. After investigating the impact of these bias words on the performance of our system (by replacing and removing them and checking the performance before and after), we combine this methodology with existing SOTA attribution techniques to verify whether they reveal similar patterns.

2 Related Research

As of late, transformer models have become an integral part of most state-of-the-art systems for irony detection. This can be done either through direct fine-tuning (Ángel González et al., 2020) or by using the contextual embeddings from a transformer model as input for a different neural classifier, like a Convolutional Neural Network (CNN) (Ahuja and

Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, pages 315–324
July 14, 2023 ©2023 Association for Computational Linguistics

Sharma, 2022). In some cases, this transformer representation input was enriched with additional features, as Cignarella et al. (2020) did by adding syntactic n-gram features. For Dutch specifically, transformers have already been fine-tuned and (quite favorably) compared to feature-based models with a focus on modeling the implicit sentiments in ironic statements (Maladry et al., 2022a).

The primary downside of these neural representation-based systems is that they are notoriously hard to interpret in a reliable way (Ghorbani et al., 2019). Nevertheless, plenty of attempts were made to explain such models. Popular techniques to gain insights into transformer models can be classified in two groups: (1) feature perturbation and (2) attention-based approaches. Most perturbation approaches, such as SHAP (Lundberg and Lee, 2017) and LIME (Ribeiro et al., 2016), are model-agnostic techniques that compare how changes in the input (feature representation) affect the output (prediction). While the aforementioned systems ignore the underlying model architecture, Integrated Gradients (IG) (Sundararajan et al., 2017) keeps track of the gradient changes caused by gradually lowering the feature values so as to find the features that arouse important gradient changes. As this makes it a neural-specific perturbation approach, it is positioned closer to the system architecture. One variation, called Discretized Integrated Gradients (DIG), grounds the perturbations in representations of sub-words that exist in the vocabulary of a transformer's tokenizer (Sanyal and Ren, 2021). By using real anchor-words with (mostly) lower feature values as interpolation points, this technique creates a non-linear path towards the padding-token (where all feature values are 0).

The model-specific alternative (2) to perturbation approaches instead relies on the attention weights and estimates the feature importance by tracking the activations in the neural network for a single input. Though popular, the use of attention weights as an explanation has been the topic of heavy discussion, with researchers opposing (Jain and Wallace, 2019) and supporting (Wiegreffe and Pinter, 2019) their reliability. The most popular explanation only accounts for the attention values in a single (usually the final) layer (Xu et al., 2015). More refined versions of this approach estimate the feature importances by combining the attention weights of different layers, either through averaging or with more sophisticated methods such as attention roll-out (Abnar and Zuidema, 2020) and layer-wise relevance propagation (LRP) (Montavon et al., 2017). While the use of attention is especially popular in computer vision, the more advanced methods have been successfully implemented and validated for transformer models for NLP by Chefer et al. (2021), who showed that their improved implementation of LRP gives better class-specific explanations compared to roll-out because the latter tends to attach too much importance to irrelevant tokens. Further mentions of LRP in this paper follow the implementation of Chefer et al. (2021).

To our knowledge, model explainability has not been researched thoroughly for irony detection. For Dutch, a manual analysis of wrong predictions already reveals some limitations of SOTA systems, including transformers (Maladry et al., 2022b). One of these limitations is a strong reliance on formulaic expressions and the hypothesis that (intense) positive sentiment is prone to function as an irony trigger. Still, this remains largely based on intuition and does not include an extensive investigation with existing attribution techniques for explainability.

3 Investigation of Potential Bias Words

3.1 Data description

For our experiments, we focus on a model that is fine-tuned on a data set for Dutch irony detection (Van Hee et al., 2016). This balanced data set contains 4453 train samples and 1113 held-out test samples. The ironic tweets in this corpus were collected using the Twitter API with irony hashtags, such as #ironie, #sarcasme and #not as search terms. These irony hashtags were then removed for training and testing. The non-ironic tweets were collected from the same users that wrote the ironic tweets and were then manually annotated to make sure that they are not ironic. All tweets in this corpus were labeled using a fine-grained annotation scheme, which also includes subcategories of irony, such as *irony by clash*, *situational irony* and *other verbal irony*. However, we will not elaborate on the distributions of these labels, as we only use the binary irony labels for this research.

bias word	EN	corr.	category
goed	good	0.106	adjective
school	school	0.0938	topical noun
weer	again	0.0906	intensifier
fijn	nice	0.0883	adjective

Table 1: A short list of the most correlated lemmatized unigrams in the data set with translation, Matthews' correlation to the binary irony label and the manually assigned category. The complete table with all n-grams and correlations is publicly available at `https://github.com/aMala3/DataBiasForIrony`.

3.2 Identifying Bias Words

Our first goal is to detect “bias words”. In this research, "bias" is defined as spurious associations between word n-grams and class labels in our data set (Van Hee et al., 2016) that can cause the system to rely too much on superficial cues and miss the essence of the irony. For example, if many ironic tweets happen to contain a positive sentiment word, this could cause the system to use intense positive sentiment as a proxy for irony, as is suggested in Maladry et al. (2022b). To verify this hypothesis, we first split the sentences into lemmatized n-grams and use the presence of each n-gram as a binary feature. Subsequently, we calculated the correlation between the binary n-gram features and the irony label across the train set using Matthews' Correlation (also known as *phi coefficient*), a form of Pearson Correlation adapted for two binary values (being presence of the lemmatized n-gram on the one hand, and the irony label on the other hand).

We found that most correlated unigrams (potential spurious biases) can be classified into syntactic and semantic categories (as illustrated in Table 1). For tokens with a positive correlation, we identified the following categories: adjectives with a positive sentiment, intensifiers (including interjections and exclamations used to intensify the overall expression), and topical nouns. This final category includes nouns like “school”, “exam”, “train” or “bus” that have a strong semantic connection to a specific topic, such as *education* and *public transport*, respectively. On the other side of the spectrum, some of the tokens with a strong negative correlation include hyperlinks (to images, websites, etc.) and laughing, smiling or heart emojis and emoticons such as <3, :) and ;).

For larger n-grams, i.e. bigrams and trigrams, we found that they often include an important unigram and one or more common collocations, like “zo goed” (EN: so good), “wat fijn” (EN: how nice). In other cases, they can be a part of a complete formulaic expression like “goed begin van de dag” (EN: good start to the day). Since this means that there is a significant overlap between the different n-grams, we focus on the unigrams for this study.

system	lab.	prec.	rec.	F1	N
Full	0	0.74	0.74	0.74	2245
	1	0.74	0.74	0.74	2208
Adj.	0	0.69	0.55	0.61	164
	1	0.84	0.91	0.87	431
Int.	0	0.66	0.60	0.63	472
	1	0.77	0.82	0.79	778

Table 2: System performance of the fine-tuned RobBERT model in 10-fold cross-validation on the train set. We compare the performance on the complete train set (Full) to the performance on the subsets where all tweets contain one of the identified adjectives (Adj.) or intensifiers (Int.).

3.3 Quantitative Analysis on Subsets

To investigate whether the correlation between those potential bias words and the irony label is not groundlessly used as an approximation of irony, we evaluate the performance of a fine-tuned transformer on tweets that specifically contain these words.[1] We focus on the first two categories (adjectives and intensifiers) as they can be more easily isolated. For evaluation, we fine-tuned the pretrained Dutch RobBERT (Delobelle et al., 2020) model in a 10-fold cross-validation setting. All systems for this paper were fine-tuned for 2 epochs (200 warm-up steps) with an AdamW optimizer, learning rate of 5e-5, weight decay of 0.01, evaluating every 200 steps on a batch size of 8 on a NVIDIA Tesla V100 GPU with 10% of the train data (for each fold) held out as validation set.

As shown in Table 2, the system performs better for the subsets containing adjectives (*Adj.*) and intensifiers (*Int.*) than on the full set for the irony label ‘1’, with recall scores of 0.91 and 0.82. The lowest scores in the table are the recall scores of 0.55 and 0.60 on the non-ironic label. The combination of these two findings indicates that the system overgenerates irony predictions and has some room for improvement on non-ironic texts in the subsets for adjectives and intensifiers.

[1] We limited this list to the 50 most correlated words and manually verified them.

In a next step, we wanted to investigate to what extent the potential bias words serve as trigger words and are therefore responsible for overgenerating irony predictions. We therefore systematically adapt the input by changing the potential bias words and examine how this affects the performance on these subsets.

4 Modified Samples

4.1 Modification

The first category of potential bias words we investigate contains adjectives that are generally used to explicitly express positive sentiment about a situation with a negative connotation. As this is essential to the irony, the adjectives cannot be simply removed. Instead, we propose replacing sentiment words with a strong correlation to the irony label with a synonym that has a weaker or no correlation. To find synonyms for the adjectives, we generated the 20 nearest neighbors using fastText embeddings (Bojanowski et al., 2016) from a pretrained Dutch model with an embedding size of 300 (Grave et al., 2018). The downside of using such a similarity-based approach is that that the proposed "synonyms" can have the opposite sentiment compared to the original adjective. For the word "gelukkig" (EN: lucky or luckily), for example, one of the suggested synonyms was "helaas" (EN: unfortunately) and for "goed" (EN: good), the most similar word was "slecht" (EN: bad). To ascertain the validity of our irony label, as well as the semantic and structural integrity of our sentences, we manually verified and selected a list of relevant synonyms for each adjective. In this case, we employed two setups: one always opting for the least correlated word in the (automatically generated but manually verified) synonym set and one randomly selecting one of the possible synonyms (that are not ranked in the top 50 most correlated lemmas but can still have some correlation to the label).

The second category contains intensifiers (including interjections and exclamations), which function as a supporting element for the expression of irony. As these are the building blocks for hyperboles and exaggerations, they are very common and help clarify that a message should be interpreted as ironic. However, they are generally not essential to recognize the irony because they only intensify the contrasting sentiment that is already expressed by the other words in the sentence. Therefore, we assume they can be removed without significantly altering the meaning of the sentence or flipping the irony label.

set	lab.	prec.	rec.	F1	prob.
adj.	0	0.69	0.55	0.61	0.252
OG	1	0.84	0.91	0.87	0.748
adj.	0	0.64	0.55	0.59	0.262
corr.	1	0.84	0.88	0.86	0.738
adj.	0	0.66	0.55	0.60	0.257
rand.	1	0.84	0.89	0.86	0.743
int.	0	0.66	0.60	0.63	0.357
OG	1	0.77	0.82	0.79	0.643
int.	0	0.61	0.67	0.64	0.435
rem.	1	0.79	0.74	0.76	0.565

Table 3: System performance of the fine-tuned RobBERT model in 10-fold cross-validation on the train set. We compare the results for the subsets where all tweets contain one of the identified adjectives (adj.) or intensifiers (int.) in original form (OG) to the same subsets in modified form (corr., rand. or rem.). The last column presents the average probability of both labels in the subset.

4.2 Quantitative Analysis on Modified Subsets

As presented in Table 3, replacing the adjectives by a synonymous word only causes minimal changes in system performance across both labels. On average, replacing one of the highly correlated adjectives with the least correlated synonymous (positive) adjective (Adj. corr.) causes the irony probability to drop by 1%. When using a random synonym from the list, the average probability only drops by 0.5% (adj. rand.). This tells us that the system is well able to overcome the lexical feature level and properly generalizes the positive meaning of the adjectives.

Although replacing the adjectives only has a minimal impact, the experiments for intensifiers show different results. As is shown in Table 3, removing intensifiers (Int. rem.) from non-ironic tweets improves the recall on those tweets by 7%. By contrast, the same modification caused the recall for the irony label to diminish with 8%. Along with the average predicted irony probability dropping by 8%, this further supports the hypothesis that the system relies on positive sentiments and considers sentiment intensity a trigger for irony.

As we mentioned before, exaggerations and hyperboles may indicate irony and are therefore also an intuitive cue for humans trying to identify irony.

However, prototypical examples of explicit irony are ironic due to the contrast between the expressed and expected sentiment towards a situation, and do not solely depend on the expression of a positive sentiment or its intensity. If an automatic system is too dependent on this element that supports the expression of irony, it is likely to miss more subtle cases and mistake genuinely (intensive) positive sentiment for irony.

5 Explainability Metrics

To further solidify our understanding of how our system models irony, we also employ explainability metrics that account for the system architecture and the mechanisms that drive its decision-making.

5.1 Explaining the Metrics

In our analysis, we include three metrics to locate trigger words: Discretized Integrated Gradients (DIG)[2], Layer-Integrated Gradients (LIG)[3] and an improved implementation of Layer-wise Relevance Propagation (LRP)[4], each with increasing reliance on the model architecture.[5]

DIG and LIG are two feature perturbation approaches based on Integrated Gradients. This means that they estimate feature importance by using alternative input representations (perturbations) with gradually lowered feature scores and comparing those feature changes to the resulting gradient changes. Unlike the original implementation of Integrated Gradients, which uses linear paths to scale down the feature representations, DIG creates a non-linear interpolation path by sending it through the representations of *real* anchor words. These anchor words are sub-words that exist in the vocabulary of the transformer model's tokenizer. LIG, on the contrary, does not account for real word representations, but instead considers how the activations in the final layers are influenced by the activations in previous layers. Compared to those approaches, the LRP attribution technique relies completely on the activations, weights and biases triggered by the current feature representation and does not consider alternative input representations. This implementation (Chefer et al., 2021) uses Deep Taylor Decomposition to attribute importances in each layer, which are then propagated backward throughout the network to result in total attributions for (only) the predicted label.

5.2 Setup of the Analysis

All applied explainability metrics assess how each sub-word in a text impacts the final prediction. Therefore, the resulting attributions are only valid on a local level (i.e. single text samples) and are not general model features. To overcome this issue, we perform both a manual analysis on the local tweet level and a search for generally relevant tokens.

For our manual evaluation, we investigated a random sample of texts with a focus on the sub-sets discussed in Sections 3 and 4. This manual analysis consisted of two parts: first, we compared the different metrics among each other (on the same sample of 50 tweets for all 3 metrics) and second, we looked for systematic attribution patterns (on an additional sample of 100 tweets). To estimate general feature importances in a more quantitative way, we calculated the average attributions for each subtoken in the complete train corpus[6]. This allows us to verify which sub-words our system generally considers more important and complements our manual analysis.

In Section 5.3, we discuss the first part of our manual analysis (i.e. comparing the different metrics). Subsequently, in Section 5.4, we combine the insights from the second part of the manual analysis with an inspection of average token attributions to discuss the general attributions patterns.

5.3 Comparing Explainability Metrics

To gain intuitive insights in the respective quality of the different explainability metrics, we performed a manual analysis using the three metrics on a set of 50 samples, half of which containing a positive sentiment word (1) and the other half containing an intensifier (2). In the following examples (Figures 1 and 2), we present the tokens with a positive attribution (for the irony label) in green and the negative attributions (for the non-ironic label) in red, with brighter colors presenting stronger attributions. For DIG and LIG, a sentence can contain both positive and negative attributions, but this is not the case for LRP as the attributions are label-specific. For each of the metrics, the visualizations were generated with Captum (https://captum.ai/), a

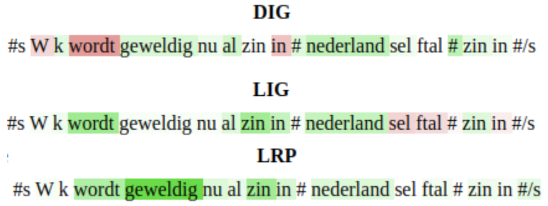
[2] https://github.com/INK-USC/DIG

[3] https://github.com/cdpierse/transformers-interpret

[4] https://github.com/hila-chefer/Transformer-Explainability

[5] For each text, we use the explanation for unseen data, meaning that the attributions were generated with the same 10-fold cross-validation setup we used in Section 3.3.

[6] https://github.com/aMala3/DataBiasForIrony


DIG

#s W k wordt geweldig nu al zin in # nederland sel ftal # zin in #/s

LIG

#s W k wordt geweldig nu al zin in # nederland sel ftal # zin in #/s

LRP

#s W k wordt geweldig nu al zin in # nederland sel ftal # zin in #/s

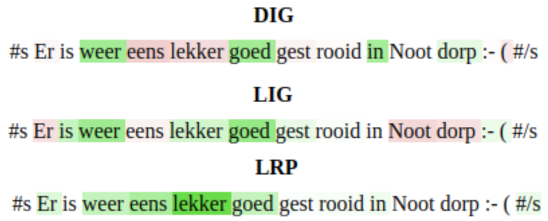
Figure 1: World Cup is going to be great, already looking forward to it #DutchTeam #lookingforwardtoit

specialized library for visualizing explanations for neural networks.

After manual inspection, we found that DIG provides the least intuitive importance attributions, whereas LIG and LRP seem to work better for irony detection. In examples 1 and 2, “wordt” (EN: will be) and “in” (EN: in) gain strong attributions, even though these words only serve a functional purpose. This issue, where high importance is assigned to irrelevant tokens such as non-creative punctuation (e.g., a comma or full stop) and function words (e.g. articles), seems to be the most common for DIG, less common for LIG and the least common for LRP. For DIG, we found that non-zero attributions were assigned to irrelevant punctuation in 18 out of 50 samples (36%) and to irrelevant function words in 26 samples (52%). Meanwhile, for LIG we observed irrelevant punctuation in 7 cases (14%) and irrelevant function words in 17 cases (34%). For LRP, irrelevant punctuation was only overestimated in 3 cases (6%) and function words in 7 cases (14%). Whereas the DIG (and to a lesser extent LIG) attributions for “weer eens lekker goed” (EN: really did a great job again) vary between positive and negative, LRP attributions recognize them as a single span and the primary reason for the irony prediction in this tweet. On the same 50 samples, the DIG attributions gave such opposite attributions within a single word (split into sub-words) or text span in 30 samples (60%), as opposed to the LIG attributions, which only showed this in 18 samples (36%). This issue cannot occur for LRP, because it has label-specific attributions. To conclude, since we showed that LRP is less likely (1) to attribute importance to irrelevant tokens, and (2) to attribute contradicting importances to sub-tokens that belong together, LRP revealed to be the most meaningful metric for our analysis.

5.4 Discussing the Attribution Patterns

Combining our extensive manual analysis on a larger sample (100 additional tweets) with an inspection of the averaged token attributions allows us to confidently present the following insights. First, we found that the intensifiers, interjections and exclamations indeed receive high attributions, especially when combined with positive sentiment words. Those positive sentiment words, like “goed” (EN: good) and “fijn” (EN: nice) also receive relatively high attributions by themselves. A manual check of the sub-tokens with an average attribution of over 0.75 revealed that 111 of the 254 subtokens (44%) have a positive sentiment. Replacing these sentiment words by synonyms (as was done in Section 4) with a more intense sentiment, such as “geweldig” (EN: great) and “fantastisch” (EN: fantastic), barely increases the attributions. Still, adding a lexical intensifier to a positive adjective results in a larger increase. The highest attributions are often linked to either the intensifier or the positive adjective, without any clear reason to choose one over the other. When several intensifiers and sentiment words co-occur in the same text, the attribution methods stack the attributions on one or a few words, while disregarding the others. As shown in Figure 3, LRP correctly spreads the attributions evenly across a typical formulaic expression.


DIG

#s Er is weer eens lekker goed gest rooid in Noot dorp :- (#/s

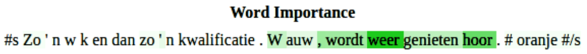
LIG

#s Er is weer eens lekker goed gest rooid in Noot dorp :- (#/s

LRP

#s Er is weer eens lekker goed gest rooid in Noot dorp :- (#/s

Figure 2: They really did a great job again sprinkling road salt in Nootdorp :-(

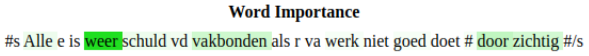

Word Importance

#s Zo ' n w k en dan zo ' n kwalificatie . W auw , wordt weer genieten hoor . # oranje #/s

Figure 3: This kind of World Cup and then this kind of qualification. Wow, will be fun again. #orange

Some topics with high correlations in Section 3 also achieve high LRP attributions on average. This is shown in the topic *politics*, represented by subwords like “conservatieve” (EN: conservative; with an average attribution of 0.95).

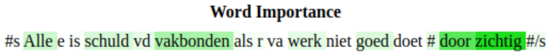
However, this is not the case for *“school”*, related to the topic *education*), and *trein* (EN: train), related to the topic*public transport*). Based on the lemmatized unigram correlations, “school” has the second highest correlation with the irony label, but



Word Importance

#s Alle e is weer schuld vd vakbonden als r va werk niet goed doet # door zichtig #/s

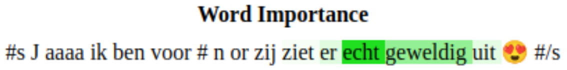
Figure 4: The *unions* are to blame for everything *again* if rva does not do its job properly *#transparent*



Word Importance

#s Alle e is schuld vd vakbonden als r va werk niet goed doet # door zichtig #/s

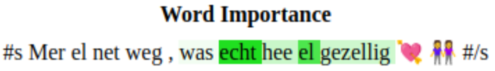
Figure 5: The *unions* are *to blame* for *everything* again if rva does not do its *job properly #transparent*



Word Importance

#s J aaaa ik ben voor # n or zij ziet er echt geweldig uit 😍 #/s

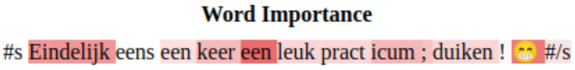
Figure 6: Yaaa, I'm here for #nor she looks *really great*



Word Importance

#s Mer el net weg , was echt hee el gezellig 💘 👫 #/s

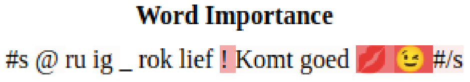
Figure 7: Merel just left, was *really veery nice*



Word Importance

#s Eindelijk eens een keer een leuk pract icum ; duiken ! 😁 #/s

Figure 8: Finally a fun particum; diving! :D



Word Importance

#s @ ru ig _ rok lief ! Komt goed 💋 😉 #/s

Figure 9: #ruig_rok sweet! will be all right

the sub-token "school" only has an average LRP attribution of 0.07. Likely, this is due to the fact that the attributions tend to relate to intensifiers and positive adjectives. We assume this because the manual evaluation revealed that removing intensifiers can occasionally redirect the attributions to topics, as shown in Figure 4 (before removal) and Figure 5 (after removal). Here,"weer" (EN: again) is the intensifier, and "vakbonden" (EN: labor unions) are the topic of the evaluation.

Surprisingly, the system also attributes high scores to nouns that do not fit the expected topical nouns, but that instead have a strong negative connotative sentiment, such as "wereldoorlog" (EN: world war) and "dictatuur" (EN: dictatorship). Alongside the aforementioned topical nouns, this could indicate that the system already models some connection between positive (adjectives) and negative (topics) sentiments within the same sentence. This is also visible on a larger scale, when comparing the average attributions on the positive adjective subset to the same subset where the adjectives are replaced by less correlated synonyms. The average attribution of "tandarts" (EN: dentist) changes from 0.69 to 1 when replacing a highly correlated adjective with a less correlated synonym, as was done in Section 4. Still, this currently seems to be limited to very popular topics for irony and nouns with a strong negative sentiment.

When a text contains intense positive sentiment, but is not intended in an ironic way, it is at risk to be mistaken for irony. As shown in Figures 6 and 7, both the intensifiers, "echt" (EN: really) and "heel" (EN: very), as well as the positive adjectives "geweldig" (EN: amazing) and "gezellig" (EN: pleasant / cozy), achieve high attributions for the irony label. Notably, the positive emojis do not receive any attributions for the irony label, while they carry the same sentiment. In fact, when looking at the correctly classified genuinely positive texts (Figures 8 and 9), this type of positive emoji serves as a trigger for the non-ironic label. Based on the averaged token attributions[7], this generally seems to be the case for a selection of positive emojis. Moreover, the correlation between single tokens and the irony label in Section 3 already identified this as a potentially spurious bias.

Altogether, these results suggest that, while the performance of transformers for irony detection is quite good, the way our system models irony remains rather superficial since it seems to depend on the detection of lexical exaggerations of positive sentiment. Although there are some indications to argue that the model can partially model the contrast that is so essential to irony, the system seems to only pick up the most extreme contrasts and most common topical nouns. For the non-ironic label, the system has also modeled an exaggerated relation between the use of positive emojis and a text being sincere because they were simply more common in those texts. In the end, the patterns in our transformer model are more similar to the simple correlations calculated in Section 3 than expected.

6 Training with Modified Data

As argued in Section 4, the modified tweets can keep the same irony label as the original tweets. This means that they could also prove helpful as additional train data, as they introduce (1) more lexical variety to the data set by using less frequent synonyms and (2) are also examples of irony with a

[7] Available at https://github.com/aMala3/DataBiasForIrony.

system	prec.	rec.	F1	acc.
Base	0.7242	0.7210	0.7210	0.7227
Aug	**0.7245**	**0.7221**	**0.7222**	**0.7237**

Table 4: System performance of the fine-tuned Roberta models on the held-out test set. Scores are averaged over 5 train runs to overcome the relatively difference between highest and lowest F1-score (2-3%) depending on the seed.

lower sentiment intensity. Therefore, the modified samples could help improve the robustness of our model and allow it to recognize more subtle ironic expressions.

In Table 4, we present the scores on the held-out test set for the two versions of the train set: (1) the original train set (*Base*) with a train size of 4007 and validation size of 446 and (2) the original train data including the modified samples (*Aug*) with a train size of 5794 and validation size of 645 (a total increase of 1986 samples).

While the augmented system may be more robust in practice, fine-tuning RobBERT on (2) the data set with modified examples shows no direct improvement over the original train set when evaluating on the held-out test set. We hypothesize this may be related to the fact that the same biases that are part of the train data are also present in the test data, as it is part of the same data set.

7 Conclusion

In this paper we investigated the origin and effect of bias on automatic irony detection in Dutch. By looking into the potential biases that could emerge from the data, we found that these can be classified into three categories: (1) positive sentiment words (mostly adjectives), (2) intensifiers, interjections and exclamations and (3) topical nouns. To investigate whether our fine-tuned transformer model uses these biases as trigger words, we evaluated the system performance on subsets that specifically contain those bias words (i.e. positive sentiment words and intensifiers) and compared the results to modified samples where the adjectives were replaced by synonyms and the intensifiers were removed. In addition, we also investigated how our system models irony using three state-of-the-art explainability techniques that assign feature attributions to each of the sub-tokens in a text: Discretized Integrated Gradients (DIG), Layer Integrated Gradients (LIG) and Layer-wise Relevance Propagation(LRP). After generating LRP attributions for every text in our train set, we ranked the different sub-tokens according to their average impact on the prediction.

Both of these methods support the hypothesis that our fine-tuned system for irony detection strongly relies on positive sentiment and is particularly triggered by intense positive sentiment. Although some of the common topics of ironic tweets are partially recognized as important and become slightly more important when the intensifiers are removed, the system generally does not pay too much attention to them.

While intense positive sentiments are most commonly used in hyperboles or exaggerations, which is a rhetorical device used to support the expression of irony, they are not the sole solution for irony detection. As shown in our analysis, more subtle irony can go undetected and genuinely positive texts are often wrongly classified as ironic. Therefore, we attempted to use modified samples with less intense sentiments to augment our train data. Although there was no noticeable increase in performance in the held-out test set, further testing on new external data sets is needed to make reliable conclusions.

For future research, we suggest further investigating how these biases could be mitigated to make sure genuine sentiment is not mistaken for irony. This could either be done by further augmenting the data or by adapting the model. A data-driven approach we propose is to create counterfactual samples, where an ironic tweet is made non-ironic and the other way around. This is, however, no simple feat due to the creativity in ironic expressions. Likewise, our approach for creating modified samples can still be improved to result in a fully-automatic framework for data augmentation. As model adaptation, we propose improving the system with additional features that, for example, represent relevant common-sense knowledge. Finally, it would also be interesting to use human annotations of trigger words for irony detection and compare the perspectives of annotators to the model explanation.

Acknowledgements

This work was supported by Ghent University under grant BOF.24Y.2021.0019.01 as well as the interdisciplinary NewsDNA project under grant BOF.GOA.2018.0006.03.

References

Samira Abnar and Willem Zuidema. 2020. Quantifying attention flow in transformers. *arXiv preprint arXiv:2005.00928*.

Ravinder Ahuja and Subhash Chander Sharma. 2022. Transformer-based word embedding with cnn model to detect sarcasm and irony. *Arabian Journal for Science and Engineering*, 47(8):9379–9392.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Hila Chefer, Shir Gur, and Lior Wolf. 2021. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 782–791.

Alessandra Teresa Cignarella, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, Paolo Rosso, and Farah Benamara. 2020. Multilingual irony detection with dependency syntax and neural models. *arXiv preprint arXiv:2011.05706*.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. RobBERT: a Dutch RoBERTa-based Language Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3255–3265, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Amirata Ghorbani, Abubakar Abid, and James Zou. 2019. Interpretation of neural networks is fragile. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3681–3688.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. *arXiv preprint arXiv:1902.10186*.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Aaron Maladry, Els Lefever, Cynthia Van Hee, and Veronique Hoste. 2022a. Irony detection for dutch: a venture into the implicit. In *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 172–181.

Aaron Maladry, Els Lefever, Cynthia Van Hee, and Véronique Hoste. 2022b. The limitations of irony detection in dutch social media. *Language Resources and Evaluation*.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. 2017. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern recognition*, 65:211–222.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Soumya Sanyal and Xiang Ren. 2021. Discretized integrated gradients for explaining language models. *arXiv preprint arXiv:2108.13654*.

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. 2016. Exploring the realization of irony in twitter data. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1795—-1799, Paris, France. European Language Resources Association (ELRA).

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. *arXiv preprint arXiv:1908.04626*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR.

José Ángel González, Lluís-F. Hurtado, and Ferran Pla. 2020. Transformer based contextualization of pre-trained word embeddings for irony detection in twitter. *Information Processing & Management*, 57(4):102262.

8 Limitations

The primary limitation for this work is the difficulty of telling the difference between mistakes made by automatic systems and wrongly assigned importances from the attribution techniques. Additionally, our work currently only relies on a single pre-trained model that was fine-tuned on the currently only available data set for Dutch irony detection. The patterns in computational modeling we described only apply to this particular system and data set and may very well differ when training a different model on data that was collected in a different way, where the data set may rely on different patterns and biases. Finally, despite the good agreement scores (a Cohen's kappa of 0.84) for binary irony classification, this remains a complex task where annotators can be uncertain about the label. In the end, the annotators for any irony or sarcasm detection task can only make assumptions about what the author of a text intended to convey. For our setup, both the annotators and the automated system predict whether a text is ironic without considering the corresponding context. In a realistic setting, most social media texts are reactions to previous comments or external events that can be essential in order to recognize the irony. This means that the model predictions can differ from the annotated label but still be a plausible interpretation.