Irony Detection for Dutch: a Venture into the Implicit

Aaron Maladry, Els Lefever, Cynthia Van Hee, Veronique Hoste
LT3, Language and Translation Technology Team,
Ghent University, Belgium
firstname.lastname@ugent.be

Abstract

This paper presents the results of a replication experiment for automatic irony detection in Dutch social media text, investigating both a feature-based SVM classifier, as was done by Van Hee et al. (2017) and a transformer-based approach. In addition to building a baseline model, an important goal of this research is to explore the implementation of common-sense knowledge in the form of implicit sentiment, as we strongly believe that common-sense and connotative knowledge are essential to the identification of irony and implicit meaning in tweets. We show promising results and how the presented approach can provide a solid baseline and serve as a staging ground to build on in future experiments for irony detection in Dutch.

1 Introduction

Irony is traditionally defined as a rhetorical device where an evaluative utterance expresses the opposite of what is actually intended (Camp, 2012; Burgers, 2010; Grice, 1978b). In order to understand the intended implicit meaning of such an ironic utterance, the message often requires presupposed common-sense knowledge. We expect most people to know that 'walking in the rain' is not pleasant or that 'visiting the dentist' can result in a painful experience. In addition to such presupposed knowledge, a sarcastic or ironic utterance is often enriched with an explicit mention of the opposite sentiment [1]. When using sarcasm[2], we usually are not just 'fine' with walking in the rain but we say we 'love it'. People are able to recognize such figurative language because we possess both the common-sense knowledge and can catch explicit semantic or lexical cues. Previous research has proven the value of lexical and semantic features for irony detection and has shown that they already allow us to recognize some cases of irony (Cignarella et al., 2020; Van Hee, 2017).

Gathering and modeling the common-sense knowledge required for irony recognition is more problematic. How can we expect an automatic system to know that an implicit negative connotation is attached to an event expressed in a given utterance, if the exact opposite information is provided in a text like 'Oh god, I love it when I have to walk home in the rain!'. In our research, we aimed to identify the general implicit sentiment behind a concept or event by looking at the tweets other people have posted containing that very concept or event. If 9 out of 10 people complain about 'walking in the rain', people who say they 'love it' might very well say that ironically. The combination of lexical features and this kind of data-driven common sense are the foundation of our machine learning approach for irony detection, which has already been applied successfully to English data (Van Hee, 2017). Yet, transferring the methodology from a high-resource language (English) to a lesser-resourced language (Dutch) is not as straight-forward as it might seem. English language models and lexicons can generally rely on larger amounts of data, and not every previously-used resource for the task (e.g. SenticNet (Cambria et al., 2020)) includes Dutch data or has a Dutch counterpart. In addition, the concept of irony might be language-universal, but the realization of irony might employ different language-specific tools and structures. In the next Section, we will discuss related research and the most recent approaches applied to irony detection. Next, we give a short description of the experimental corpus.

[1] This kind of sentiment clash was first examined by Riloff et al. (2013).

[2] Technically, there is a small difference between the two. Sarcasm is regarded as more negative and suggests a harshly-intended form of irony used to mock or ridicule someone. However, not only in popular speech and social media, but also in academic literature, the term 'sarcasm' is often used interchangeably with 'irony'. Therefore, we take note of the negative connotation but use the terms as synonyms, as is done in the related research (Van Hee et al., 2016a; Filatova, 2012; Jijkoun and Hofmann, 2009)

Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis, pages 172 - 181

Section 4 follows an elaborate description of the proposed systems and an explanation of the features we developed for our classifiers. Finally, we present the results of our experiments and we wrap up the paper with a conclusion and identify some avenues for future research.

2 Related Research

The detection of sarcasm and irony remains one of the primary hurdles for sentiment analysis and other natural language processing (NLP) tasks like detection of cyberbullying, humor and toxicity.

When it comes to methodology, the techniques applied to English irony detection range from feature-based classifiers to neural networks and transformers, including many combinations of (transformer-generated) embeddings with neural or traditional classifiers. Feature-based classifiers like Support Vector Machines or Classifiers (SVM or SVC) are flexible and can easily be equipped with new features, but they generally require a lot of manual work. Van Hee (2017) provides a successful example of an approach combining lexical, semantic and syntactic features for a strong baseline system. This kind of supervised classifier is advantageous when determining the informative value a specific linguistic feature (like part-of-speech tags) contributes to the decision-making process.

Transformer language models (Devlin et al., 2019) and bidirectional transformers (Vaswani et al., 2017) currently occupy the throne of the state-of-the-art in NLP. While these language models are remarkably adaptable and perform well for a variety of tasks (providing they have been fine-tuned for classification), they usually do not suffice on their own. Often the language models are used to generate word embeddings as a (partial) input for a traditional or neural classifier. Potamias et al. (2020) combine the embeddings from RoBERTa, a robust bi-directional transformer approach, with a recurrent convolutional neural network. Cignarella et al. (2020) use a Long Short-Term Memory (LSTM) neural network to exploit both transformer (BERT) word embeddings and syntactic dependency-based n-gram features (Sidorov et al., 2012).

In the SemEval 2018 shared task for irony detection (Van Hee et al., 2018b), the best-performing model (Wu et al., 2018) exploited word embeddings, syntactic and sentiment features using a Long Short-Term Memory neural network. Other participants made use of ensemble learning techniques with majority voting on neural network approaches. One example is presented by Baziotis et al. (2018), who used ensemble learning of two LSTM models, one exploiting character n-grams and the other word n-grams. The third ranking approach also used ensemble learning, but instead opted for traditional machine learning classifiers (Logistic Regression and Support Vector Machines) with word embeddings and manually extracted features (Rohanian et al., 2018).

While research into English sarcasm and irony detection is thriving and receives a lot of attention, other languages are lagging behind (Cakebread-Andrews, 2021). SemEval 2022 again includes a sub-task specifically for sarcasm detection in English and Arabic (iSarcasmEval), which aims to both improve the state-of-the-art methodology for English and expand the scope to less-researched languages. Irony detection for Dutch is still in its infancy. Ever since Kunneman et al. (2015) and Van Hee et al. (2016a) collected and analyzed Dutch irony corpora, and gathered some initial insights into irony detection for Dutch, no new research on the topic has been presented (to the best of our knowledge).

One way to overcome the lack of language-specific research is the use of multilingual language models, like multilingual BERT (Devlin et al., 2018), which makes sense as irony and sarcasm are assumed to be language-universal. Multilingual approaches that utilize the exact same feature set and language models, have shown promising results, but generally do not aim to outperform language-specific models and rather attempt to catch up to their performance levels. An example is the language-universal approach presented by Cignarella et al. (2020), who successfully created a syntactically informed BERT model for English, French, Italian and Spanish social media data. Dutch was not included in this research.

3 Experimental Corpus and Data Description

For this research, we made use of the Dutch data set for irony detection collected by Van Hee et al. (2016a). The balanced corpus consists of 5,566 annotated tweets and was gathered in two ways. One part (3,179 tweets) contains irony-related hashtags (i.e. *#sarcasme, #ironie, #not*) and was annotated with a fine-grained annotation scheme. The

	irony 3-way	irony binary	hashtag indication	polarity contrast
Dutch data set	0.77	0.84	0.60	0.63

Table 1: Cohen's kappa scores for all annotator pairs as presented in Van Hee et al. (2018a).

remaining tweets, which balance out the corpus, were posted by the same users as the ironic tweets and were confirmed to be non-ironic by annotators. Out of the 3,179 tweets with irony-related hashtags, 6% were found to be non-ironic. This is notably lower than in the English data set, which was collected and labeled for SemEval 2018 using the same annotation guidelines and methods (Van Hee et al., 2018b), where 19% of the hashtag-containing tweets are non-ironic. The inclusion of "#not" as an irony-related hashtag was found to make the ironic data set less noisy for Dutch compared to English, where 'not' had sometimes been used as a negating particle rather than an irony marker.

There are a couple of reasons why we selected this data set for our research. The first and most important reason is that the tweets containing irony-related hashtags were manually checked to confirm if they are actually ironic, and to define the type of irony being used. As noted by Van Hee et al. (2017) and Kunneman et al. (2015), tweets containing irony-related hashtags can still be non-ironic and introduce noise in the corpus. Having them checked by annotators ensures a better corpus quality compared to hashtag-based approaches that are common in this research field. The fine-grained annotation guidelines are another useful aspect of the data set, because they allow for more insight and understandability in how irony is linguistically realized. Each ironic tweet receives a label indicating the type of irony: ironic with sentiment clash, situational irony or other irony. A first traditional distinction is made between *verbal* and *situational* irony. Situational irony happens when a situation fails to meet our expectations (Lucariello, 1994; Shelley, 2001). A common example of this is when firefighters have a fire in their kitchen while they are out to answer a fire alarm (Shelley, 2001). Verbal irony, here represented by the labels *ironic by clash* and *other irony*, is defined as expressions that convey the opposite meaning of what is said (Grice, 1975) and implies the expression of a feeling, attitude or evaluation (Grice, 1978a; Van Hee et al., 2016b). *Ironic by clash* occurs when a text expresses an evaluation whose literal polarity is opposite to the intended polarity. Any other forms of verbal irony are categorized as *other irony*. The distribution of the data is as follows:

1. Ironic: 2,783 instances
    - ironic by clash: 2,201 (79%)
    - situational irony: 190 (7%)
    - other irony: 392 (14%)
2. Non-ironic: 2,783 instances

Besides the type of irony used, the annotators also indicated whether or not the irony-related hashtags (#sarcasme, #ironie, #not) were essential to recognize the irony. In fact, more than half of the data set (53%) of the ironic tweets required the irony hashtag to be recognized as ironic by human annotators, as illustrated by the following examples:

- @user een gezellige moskee met hele tolerante gematigde lieden. #not (English: @user a cozy mosque with very tolerant moderate people. #not)
- Ge moogt allemaal fier zijn op uzelf. #sarcasme (English: You should all be proud of yourselves. #sarcasm)
- @user maar vanavond zat er in Het Journaal toch maar mooi een Belgische opiniepeiling, die weer heel ernstig werd geduid. #ironie

  (English: @user but tonight there was a nice Belgian opinion poll in Het Journaal, which was again interpreted very seriously. #irony)

The English equivalent of this Dutch data set is the foundation of the irony detection shared task of SemEval2018 (Van Hee et al., 2018b) and often used as one of the go-to data sets for irony or sarcasm detection (included in Cignarella et al. (2020); Potamias et al. (2020); Ahuja and Sharma (2021); Chowdhury and Chaturvedi (2021)). As the Dutch counterpart is collected, annotated in the same manner by native speakers and shows an acceptable level of agreement between the annotators with scores ranging from moderate to substantial (see Table 1), the quality of the data set should be comparable. For binary irony classification, the

inter-annotator agreement indicates almost perfect agreement, with a Cohen's kappa (Cohen, 1960) of 0.84. After removing the irony hashtags, the data set was randomly divided into a test and training split, respectively containing 20% and 80% of the total tweet count. This leaves 1113 tweets in the test set with a fairly balanced label distribution (52% ironic to 48% non-ironic).

4 System Setup and Features

The baseline Support Vector Classifier (SVC) system leverages a core set of lexical character and word n-gram features, which are augmented with more elaborate syntactic, semantic and sentiment lexicon features. Syntactic features include Part-of-Speech frequencies, temporal clash and named entity features. The temporal clash feature indicates whether two different verb tenses occur in the same tweet. Named entity features include both a binary feature (whether or not there is a named entity), and a frequency feature (counting the number of named entities in the tweet). Semantic features are binary features based on Word2Vec (Mikolov et al., 2013) clusters of semantically related words, generated from a large Twitter background corpus. Such a feature could, for example, check whether the tweet contains a word in the semantic cluster [school, dissertation, presentation, degree, classes, papers, etc.] (Van Hee et al., 2016a). Lastly, the sentiment lexicon features count the number of positive and negative token occurrences in each lexicon and take the sum of the sentiment values. We used a variety of sentiment lexicons, including the NRC Word-Emotion Lexicon (Mohammad and Turney, 2013), PATTERN (De Smedt and Daelemans, 2012), the Duoman Lexicon (Jijkoun and Hofmann, 2009), the Hogenboom Emoticon Lexicon (Hogenboom et al., 2013) and The Emoji Sentiment Ranking (Kralj Novak et al., 2015). All SVM models were trained using libsvm (Chang and Lin, 2011) to stick as closely as possible to the methodology of Van Hee (2017). In the same way, we optimized the hyperparameters of the SVM on the train set through grid search[3].

In addition to this baseline model, a Dutch transformer language model (de Vries et al., 2019), built from diverse corpora containing 2.4 billion tokens, was fine-tuned on the training data for irony detection. The methodology used for fine-tuning and deciding on the number of epochs is strongly based on the experiments of Van Hee et al. (2021), who adapted the same transformer (BERTje) for sentiment analysis on news data. The transformer model was thus trained to classify the tweets as ironic or not for 15 epochs with AdamW (Adam optimizer with weight decay) as the optimization algorithm and a learning rate of 5e-05 (Van Hee et al., 2021). The number of epochs was decided on by evaluating the F-score on a held-out validation set (10% of the train data), to keep adding epochs as the F-score improved.

Finally, we created an additional feature to add to our baseline SVC model: *implicit sentiment clash*. This feature captures a clash between the sentiment of an annotated *irony target* and an explicit mention of the opposite sentiment, which is extracted from the remainder of the tweet based on the aforementioned sentiment lexicons. The annotated *irony targets* denote the topic of the ironic utterance. In the annotation guidelines they are defined as "text spans whose implicit sentiment (i.e. connotation) contrasts with that of the literally expressed evaluation" (Van Hee, 2017). These strings can be of any length and syntactic structure, for example "group assignments" or "can't sleep".

We developed two versions of the implicit clash feature. One version utilizes the annotated sentiment of the irony target (considered the gold standard implicit sentiment) to determine the upper boundary for the integration of implicit sentiment. In other words, this scenario presumes perfectly inferred implicit sentiment for each annotated target. The other version of the feature deducts the implicit sentiment automatically with a data-driven approach. To this end, a new set of tweets was collected for each individual target string to function as a background corpus from which we derive implicit sentiment. We fine-tuned a transformer model for sentiment analysis, implementing the same methodology as was used for the inference of implicit sentiment in news data by Van Hee et al. (2021) but trained the model on our own sentiment data[4]. Automatic sentiment analysis thus determined the sentiment of each tweet in the background corpus and we then grouped the resulting sentiments per irony target. Based on

[3]For all SVM models, the optimal c and gamma values turned out to be 2 and 0.00195, respectively.

[4]This corpus contains review texts collected in the framework of a student assignment in a course on Digital Communication. The same data set was utilized for the creation of the LT3 demo for sentiment analysis (https://www.lt3.ugent.be/sentiment-demo/).

the number of positive, neutral and negative sentiments for each target, the most common of the three is assumed as the target's implicit sentiment. For the target "drilled awake", for example, 44% of the tweets were classified as negative, 22% as neutral and 33% as positive. Because the negative partition is the largest, we assign a negative polarity to that target. After determining the implicit sentiment of the target using the method described above, we looked for the presence of a sentiment clash by searching the remainder of each tweet (without the target string) for any positive or negative sentiment token in any of our sentiment lexicons (NRC, PATTERN (De Smedt and Daelemans, 2012), Duoman (Jijkoun and Hofmann, 2009), Hogenboom (Hogenboom et al., 2013) and Emoji (Kralj Novak et al., 2015)), considering it the tweet's explicit sentiment, and compared it to the implicit sentiment of the target. This method was able to cover 756 out of 939 (81%) of all annotated targets (in test and train set). In such manner, the correct implicit sentiment was predicted for 636 out of 756 targets (84%).

If the explicit sentiment in the tweet contradicts the implicit sentiment of the irony target, we call this an *implicit sentiment clash*. This feature only occurs when we were able to determine an implicit sentiment for the annotated target, meaning there are no targets for non-ironic tweets. Despite the high coverage and accurate analysis for implicit sentiment, only 16% of our ironic test tweets and 17% in the training set received the sentiment clash feature. This might seem surprisingly low considering 79% of all ironic tweets have been annotated with the label 'ironic by clash'. However, we should keep in mind that only about a third of the tweets with the label 'ironic by clash' received an annotated irony target. A closer look reveals that the use of lexicons as indicators for explicit sentiment works quite well[5].

We believe this could still be improved. In some cases, the explicit sentiment that causes the clash was annotated as part of the irony target[6].

Besides the clash between implicit and explicit sentiment, we implemented another feature to indicate a contrast among explicitly mentioned sentiments. This time, the explicit sentiments were gathered across all lexicons collectively instead of per lexicon as was done for the baseline SVM. An explicit clash occurs, for example, when a text contains a word like "lovely" and an angry emoji or other word like "disgusting". This *explicit clash* occurs in 22% of all test tweets and co-occurs with the irony label in 58% of the cases. Although this feature did not show a high information gain in the data set (0.005), we still considered it worthwhile to combine it with the implicit clash feature for our experiments.

For the evaluation of the features and SVM models, we developed separate SVM systems containing (1) the baseline feature set, (2) all mentioned features including the implicit and explicit sentiment clash and (3) the baseline feature set with the implicit sentiment clash, but without the explicit clash[7]. For each of the systems with implicit clash as a feature, we evaluated two versions of the feature: one with the automatically predicted implicit sentiment and one with the annotated implicit sentiment.

5 Experimental Results and Analysis

First of all, we noticed that all models reached F-scores above 70% (see Table 2), which was the top result for the English data set (Van Hee, 2017). The fine-tuned transformer model (BERTje) performed the worst out of all tested systems with an F-score of 73.08%. The baseline SVM system (without the implicit sentiment feature) clearly outperforms it with an F-score of 77.82%.

Our SVM system containing the automatically generated clash feature successfully leverages the implicit sentiment of irony target strings and is able to improve the baseline F-score with another percentage. This might seem a modest improvement, but we should stress that this feature was only one out of the 15,845 features and could have been 'undersnowed' by the many lexical features.

These results are further confirmed when comparing the performance of both implicit clash models. Our automatic implicit clash model without the explicit feature even slightly outperformed the model with manually annotated implicit sentiment. We hypothesize this is because of the nature of some of the annotated strings. The annotation guidelines did not include any length or format restrictions for the irony targets, which causes them

[5]In 86% of tweets with an automatic implicit sentiment, we also detected some form of explicit sentiment.

[6]The annotators were free to choose the formats of the irony targets, so the irony target strings vary in length and syntactic format.

[7]Since this feature could introduce more noise, we also develop a system without it.

	Accuracy	Precision	Recall	F-score
Baseline				
Baseline SVM	75.47	73.02	83.30	77.82
Transformer (Bertje)	72.33	73.46	72.70	73.08
SVM with clash features				
Implicit (auto)	**77.00**	74.81	**83.65**	**78.98**
Implicit (gold)	77.00	**75.20**	82.78	78.81
Implicit (auto) and explicit	76.91	74.77	83.48	78.88
Implicit (gold) and explicit	76.82	75.04	82.61	78.64

Table 2: Overview of all experimental results (metrics in %) for binary classification (irony or not). Accuracy was calculated for the full test set. F-score, recall and precision were calculated for the positive label. By *implicit clash* we mean a contrast between implicit and explicit sentiment. An *explicit clash* is a clash between two explicit sentiments.

sometimes to be exceedingly long and therefore noisy. Some of the targets already contain a sentiment clash. The most obvious explanation would be a mistake during annotation, which makes it impossible to detect a clash between the target and the remainder of the tweet. However, it could just as well be a nested clash. In that case there would be two clashes in the tweet, one inside the target and one between the target and the rest of the tweet. Others contain common sense that is strongly connected to the physical world and require the understanding of price values for certain goods or the duration of some activities, etc. Unreasonably large amounts should generally be considered negative, but there are no methods yet to explain a machine how many agents should man a station or what the appropriate price for a t-shirt is. Many of these kind of appreciations and opinions even depend on personal, geographical or cultural preferences and characteristics. Below we present some of the targets without an implicit sentiment prediction (original Dutch tweet with English translation):

- **long and noisy targets:**
  - *En als de batterij van je Random Reader op is, kan je gelijk een nieuwe halen bij jouw Rabobank*
    (English: *and if the battery of your Random Reader runs out, you can just go get a new one at Rabobank)*
- **implicit clash:**
  - *op een zonnige zondagmiddag aan je practicum werken*
    (English: *working on your practicum on a sunny afternoon*)
- **common sense clash:**
  - *Net m'n haar gestyled, zitten er nu door de regen alweer losse krullen in*
    (English: *just straightened my hair and the rain just put loose curls in it again*)
- **complex common sense:**
  - *Er vliegen 1700 privéjets met gasten naar #Davos om de #klimaatveranderingen te bespreken*
    (English:*1700 private jets with guests are flying to #Davos to talk about #climatechange*)
- **real-world common sense**
  - *€175 voor n fietsbroek en -shirt*
    (English: *€175 for cycling shorts and shirt*)

Evaluation on a subset containing only the tweets with an annotated target leads to some fascinating outcomes (see Table 3). It seems that the impact of the missing target coverage is canceled out by the fact that many of the missing targets were actually noisy and possibly reduced prediction accuracy or is caused by a minor annotation mistake. As it stands, the predictions for the implicit sentiment work out exceptionally well, which confirms our working hypothesis that we can reliably deduct implicit sentiment using a large background corpus. As we can tell by the last three rows in Table3, the addition of the explicit clash feature did not improve our results. Consequently, we deem this feature redundant and unnecessary.

A cursory manual analysis of the wrong predictions of our best system reveals that many contain a

	Accuracy	Recall	F-score
Baseline			
Baseline SVM	89.08	89.08	94.86
Transformer (BERTje)	78.74	78.74	88.10
SVM with clash features			
Implicit (auto)	90.80	90.80	95.18
Implicit (gold)	90.23	90.23	94.86
Implicit and explicit (auto)	90.23	90.23	94.86
Implicit and explicit (gold)	90.23	90.23	94.86

Table 3: Evaluation of **only** the tweets that contained an annotated irony target (metrics in %). These tweets have all been annotated as ironic by clash. F-score, recall and precision were calculated for the positive label. We do not report the precision scores as they are all 100% because targets are only present for the positive class.

more openly expressed positive sentiment. Van Hee et al. (2018a) and Kunneman et al. (2015) both noted the relevance of hyperboles and intensifiers as linguistic features for irony detection. While not all cases of very positive sentiments are sarcastic, they do seem to occur often, especially when an irony hashtag was required to identify the tweet as ironic, as illustrated by the following examples (we show the tweets without the hashtag, as they were available to our systems):

- *Gij geeft mij echt zo een goed gevoel*

  (English: *You really make me feel so good*)

- *Maar 't is echt een heel goed idee! #alzegikhetzelf :-)*

  (English: *Well it really is a very good idea! #ifidosaysomyself :-)*)

- *Wat een heerlijk weer!*

  (English: *What wonderful weather!*)

- *Het voelt zo bijzonder als mensen op je stemmen! Dank allen voor het vertrouwen. #trots #dankbaar*

  (English: *It feels so special when people vote for you! Thank you all for the trust. #proud #thankful*)

Not every hyperbole in the test set causes misclassification, though, as many examples in the test set have been classified correctly. We hypothesize this bias could be caused by the removal of irony hashtags for ironic tweets. Whilst annotators indicated that 53% of ironic tweets required an irony-related hashtag to be recognized as ironic, we deprived the tweets of that necessary hashtag but kept the irony label. By consequence, the system might have learned to conceive a very positive sentiment as a possible indicator of irony. However, further manual evaluation and further research are needed to confirm this presumption.

The SVM with automatic implicit sentiment still attains the best results when looking at the accuracy of each model per label, as shown in Table 4. Ironically, this model does not outperform the baseline SVM on the category *ironic by clash*, which was the purpose of the implicit sentiment feature. While our transformer model achieved the best results on *not ironic* tweets, the system does not attain a higher precision on the complete data set compared to the SVMs. The smallest classes in the data set reveal the Achilles heel of the transformer model: it could not detect situational and other irony very well. One could argue that these classes only represent a small portion of the irony class[8] and that neural models would be able to generalize those given a larger data set. The SVM models, on the contrary, did not need additional data and already perform well on the different types of irony. Despite the comparable precision scores, all SVM systems surpass the transformer's recall score by about 10%. This shows the value of efficient feature engineering. Thanks to our manually-selected features, the SVMs were able to capture sarcasm and irony significantly more often than the automatically derived features used by our transformer model.

[8]The *situational irony* and *other irony* classes only contribute to 6% and 15% of the irony label in the test data respectively.

	Ironic by clash (454)	Situational irony (36)	Other irony (85)	not ironic (538)
Baseline				
Baseline SVM	87.44	75.00	64.71	67.10
Transformer (BERTje)	77.75	54.12	52.78	**71.93**
SVM with clash features				
Implicit (auto)	**87.44**	**77.78**	**65.88**	69.89
Implicit (gold)	86.56	75.00	64.71	70.82
Implicit (auto) and explicit	87.22	77.78	65.88	69.89
Implicit (gold) and explicit	86.56	75.00	68.53	70.63

Table 4: Accuracy (in %) for each system per annotated label. Per class label, we also provide the frequency of the label (between brackets). The total number of test instances is 1,113.

6 Conclusion and Future Research

This paper presents a set of experiments for irony detection on Dutch tweets. The proposed SVM models obtained good classification scores, considerably outperformed our baseline transformer model (BERTje) and were able to exploit the sentiment clash feature to achieve more accurate results. For the task of irony detection, the results confirmed that feature-based approaches, although requiring a lot of effort, obtain good results and give more insight into feature relevance and possible future improvements. Although the Dutch data set has until now remained uncharted and has no comparable results yet, the applied methodology in this replication experiment has shown 7% to 9% higher F-scores compared to the English data set.

Implicit sentiment was successfully inferred for irony targets by running sentiment analysis on a large background corpus containing these targets. Our approach using sentiment lexicons for the detection of explicit sentiment to clash with our detected implicit seems to be efficient. Our feature indicating a clash between implicit and explicit sentiment has proven to be a valuable addition to the feature set, even when it can only be activated in a portion of the tweets identified as 'ironic by clash'. It is somewhat unusual that our automatic prediction for implicit sentiment achieved better results than the feature with manually annotated sentiment. Further manual analysis of the results will be necessary to better understand this discrepancy.

A brief inspection of the misclassifications has led to the presumption that our best models have recognized hyperbolic or 'exaggerated' positive sentiment as a feature. We believe this occasionally causes misclassification of very positive texts as ironic. This might be because many tweets used to have an irony-related hashtag, which was indicated as essential to detect irony by human annotators. Nonetheless, confirming this would also require more thorough analysis of the data and predictions.

We consider the results of these exploratory experiments to be insightful but we have only scratched the surface. Testing has indicated both improvements (coverage and sentiment analysis of implicit sentiment of targets) and highlighted some weaknesses. The major challenge that remains is the automatic detection of 'irony targets', the topics or concepts people are ironic or sarcastic about. Hence, we will investigate this as the subject of our future research. On top of that, our implicit clash feature was only one of the 15,000 features, which might cause it to be 'undersnowed' by the many lexical features. Therefore, we will also experiment with ensemble learning to increase the weight of this feature.

In the large scope of irony or sarcasm detection, there are still many paths to pursue. One would be the incorporation of implicit sentiment features into other systems that exploit word embeddings, in the same way as Cignarella et al. (2020) used n-gram features. Another direction is to further expand the coverage of implicit sentiment of irony targets. This can be achieved by connecting related phrases or words like surgeon - doctor - dentist when there are no exact matches. Alternatively, graph knowledge bases, such as SenticNet (Cambria et al., 2020) can be leveraged for more advanced connections between concepts and already include sentiment related to a concept. Experiments with older versions of SenticNet as a sentiment lexicon, however, did provide worse results for sentiment analysis than our data-intensive tweet-based approach (Van Hee, 2017).

References

Ravinder Ahuja and SC Sharma. 2021. Transformer-based word embedding with cnn model to detect sarcasm and irony. *Arabian Journal for Science and Engineering*, pages 1–14.

Christos Baziotis, Nikos Athanasiou, Pinelopi Papalampidi, Athanasia Kolovou, Georgios Paraskevopoulos, Nikolaos Ellinas, and Alexandros Potamianos. 2018. NTUA-SLP at semeval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive rnns. *CoRR*, abs/1804.06659.

C. Burgers. 2010. *Verbal irony: Use and effects in written discourse*. Ph.D. thesis, UB Nijmegen.

Oliver Cakebread-Andrews. 2021. Sarcasm detection and building an english language corpus in real time. In *Proceedings of the Student Research Workshop Associated with RANLP 2021*, pages 31–35.

Erik Cambria, Yang Li, Frank Z Xing, Soujanya Poria, and Kenneth Kwok. 2020. Senticnet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 105–114.

E. Camp. 2012. Sarcasm, pretense, and the semantics/pragmatics distinction. *Nous*, 46:587–634.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27.

Somnath Basu Roy Chowdhury and Snigdha Chaturvedi. 2021. Does commonsense help in detecting sarcasm? *arXiv preprint arXiv:2109.08588*.

Alessandra Teresa Cignarella, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, Paolo Rosso, and Farah Benamara. 2020. Multilingual Irony Detection with Dependency Syntax and Neural Models. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1346–1358, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Tom De Smedt and Walter Daelemans. 2012. " vreselijk mooi!"(terribly beautiful): A subjectivity lexicon for dutch adjectives. In *LREC*, pages 3568–3572.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. Bertje: A dutch bert model. *ArXiv*, abs/1912.09582.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 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. 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. *CoRR*, abs/1810.04805.

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

H Paul Grice. 1978a. Further notes on logic and conversation. In *Pragmatics*, pages 113–127. Brill.

Herbert P Grice. 1975. Logic and conversation. In *Speech acts*, pages 41–58. Brill.

P.H. Grice. 1978b. *Further notes on logic and conversation*, volume 9, pages 113–127. P. Cole, Syntax and Semantics, New York: Academic Press.

Alexander Hogenboom, Daniella Bal, Flavius Frasincar, Malissa Bal, Franciska de Jong, and Uzay Kaymak. 2013. Exploiting emoticons in sentiment analysis. In *Proceedings of the 28th annual ACM symposium on applied computing*, pages 703–710.

Valentin Jijkoun and Katja Hofmann. 2009. Generating a non-english subjectivity lexicon: Relations that matter. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 398–405.

Petra Kralj Novak, Jasmina Smailović, Borut Sluban, and Igor Mozetič. 2015. Sentiment of emojis. *PloS one*, 10(12):e0144296.

Florian Kunneman, Christine Liebrecht, Margot van Mulken, and Antal van den Bosch. 2015. Signaling sarcasm: From hyperbole to hashtag. *Information Processing Management*, 51(4):500–509.

Joan Lucariello. 1994. Situational irony: A concept of events gone awry. *Journal of Experimental Psychology: General*, 123(2):129.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Saif M Mohammad and Peter D Turney. 2013. Nrc emotion lexicon. *National Research Council, Canada*, 2.

Rolandos Alexandros Potamias, Georgios Siolas, and Andreas-Georgios Stafylopatis. 2020. A transformer-based approach to irony and sarcasm detection. *Neural Computing and Applications*, 32(23):17309–17320.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 704–714.

Omid Rohanian, Shiva Taslimipoor, Richard Evans, and Ruslan Mitkov. 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, New Orleans, Louisiana. Association for Computational Linguistics.

Cameron Shelley. 2001. The bicoherence theory of situational irony. *Cognitive Science*, 25(5):775–818.

Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández. 2012. Syntactic dependency-based n-grams as classification features. In *Mexican International Conference on Artificial Intelligence*, pages 1–11. Springer.

C. Van Hee, M. Van de Kauter, O. De Clercq, E. Lefever, B. Desmet, and V. Hoste. 2017. Noise or music? investigating the usefulness of normalisation for robust sentiment analysis on social media data. *Traitement Automatique des Langues*, 58(1):63–87.

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

Cynthia Van Hee, Orphée De Clercq, and Véronique Hoste. 2021. Exploring implicit sentiment evoked by fine-grained news events. In *Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA), held in conjunction with EACL 2021*, pages 138–148. Association for Computational Linguistics.

Cynthia Van Hee, Els Lefever, and Veronique Hoste. 2016a. Exploring the realization of irony in Twitter data. In *LREC 2016 - TENTH INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION*, pages 1795–1799. ELRA.

Cynthia Van Hee, Els Lefever, and Veronique Hoste. 2016b. Guidelines for Annotating Irony in Social Media Text, version 2.0.

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

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Chuhan Wu, Fangzhao Wu, Sixing Wu, Junxin Liu, Zhigang Yuan, and Yongfeng Huang. 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*, pages 51–56, New Orleans, Louisiana. Association for Computational Linguistics.