Personalized Intended and Perceived Sarcasm Detection on Twitter

Joan Plepi[*] †‡ and **Magdalena Buski**[*] † and **Lucie Flek** †‡
Conversational AI and Social Analytics (CAISA) Lab
† Department of Mathematics and Computer Science, University of Marburg
‡ Department of Computer Science, University of Bonn
{plepi,flek}@bit.uni-bonn.de
magdalena.buski@gmx.de
[*] These authors contributed equally to this work

Abstract

Sarcasm detection is a challenging task for various NLP applications. It often requires additional context related to the conversation or participants involved to interpret the intended meaning. In this work, we introduce an extended reactive supervision method to collect sarcastic data from Twitter and improve the quality of the data that is extracted. Our new dataset contains around 35K labeled tweets sarcastic or non-sarcastic, as well as additional tweets regarding both conversational and author context. The experiments focus on two tasks, the binary classification task of sarcastic vs. non-sarcastic and intended vs. perceived sarcasm. We compare models using textual features of tweets and models utilizing additional author embeddings by using their historical tweets. Moreover, we show the importance of combining conversational features together with author ones.

1 Introduction

Sarcasm detection is one of the most challenging NLP tasks, having an implied negative sentiment but a positive surface sentiment (Băroiu and Trăusan-Matu, 2022). Initially, early sarcasm detection systems relied on lexical and syntactic cues (Carvalho et al., 2009; Davidov et al., 2010a; Tsur et al., 2010; González-Ibánez et al., 2011; Reyes et al., 2013). However, the intended and literal meaning of the text can be interpreted differently depending on additional contextual information and on the cultural imprint of the author as well as the audience of the utterance (Ackerman, 1982; Gibbs, 1986; Dews et al., 1995; Riloff et al., 2013; Wallace et al., 2014; Bamman and Smith, 2015; Hazarika et al., 2018). One such case is the political discourse on social media, where users often utilize sarcasm and irony to express their opinion. In datasets for sarcasm detection crawled from social media like Reddit, posts from political topics, usually dominate the other topics (Davis et al., 2018; Khodak et al., 2017), hence several models have attempted to model the topic of the tweet for sarcasm detection task (Kannangara, 2018; Ghosh et al., 2020). Therefore, the effectiveness of models, predicting whether an utterance is sarcastic or not, depends not only on the choice of the model but also on the availability and quality of a high amount of labeled data (Oprea and Magdy, 2020a). The collection of such is hampered by the aforementioned challenges.

Sarcasm can be categorized into three types based on the perception of the audience and the intent of the author. The first type of sarcastic utterance is one that is not intended as sarcastic by the author but is perceived as such by the audience. The second type is an utterance that is both intended as sarcastic by the author and perceived as such by the audience. Lastly, the third type is an utterance that is intended as sarcastic by the author, but it is not perceived as such by the audience. Prior works focus on three different methods of collecting sarcastic data, distant supervision method which uses hashtags on Twitter, manual annotation, and manual collection. However, all the previous methods were able to capture only one type of sarcasm, thus limiting their ability to train models that could detect both intended and perceived sarcasm (Joshi et al., 2016; Oprea and Magdy, 2020a; Băroiu and Trăusan-Matu, 2022).

Shmueli et al. (2020) introduces a new reactive supervision method to collect sarcastic data from Twitter. This method has two advantages that address some of the issues present in previous works by relying on cues from participants in online conversations. First, it contains both types of sarcasm intended and perceived, and also additional conversational context. Our manual analysis of the data collected with this method revealed a considerable number of false positive examples due to cue tweets indicating the need for clarification rather than pointing out sarcasm. To

Proceedings of the 3rd Workshop on Computational Linguistics for the Political and Social Sciences, pages 8–18
September 22, 2023. ©2023 Association for Computational Linguistics

address this issue, we propose an extension of the reactive supervision method that improves the rate of false positives, hence the quality of the sarcastic tweets. Moreover, we collect a dataset of 35k tweets that contain both perceived and intended sarcasm and non-sarcastic tweets. In addition, we enrich the dataset with additional contextual information regarding both conversation and authorship.

The key contributions of this paper are as follows:

(1) We collect a new dataset on Twitter by extending a semi-supervised method that uses reactive supervision and provides additional contextual information.

(2) We evaluate the models using binary classification for both sarcastic vs. non-sarcastic classes and perceived vs. intended sarcasm classes.

(3) We analyze the performance of two classes of models for sarcasm detection: (i) text-only-based models that rely solely on textual features and (ii) author-contextual-based models that use author representations based on historical tweets. In addition, we also combine textual and author features with conversational features.

2 Related Work

Collection and Labeling of Sarcastic Data Previous approaches to data collection for automatic sarcasm detection can be divided into two groups: distant supervision and manual annotation (Joshi et al., 2016; Băroiu and Trăusan-Matu, 2022). One approach requires annotators to manually label whether a given utterance is sarcastic or not (Filatova, 2012), while distant supervision focuses on automatically collecting large datasets of intended sarcasm. The automatic data collection uses specific keywords to query social networks (Davidov et al., 2010b; Barbieri et al., 2014; Ptácek et al., 2014; Khodak et al., 2017). Nevertheless, the subjectivity and sociocultural dependence of perceived sarcasm (Rockwell and Theriot, 2001; Dress et al., 2008) often lead to discrepancies between intended and perceived sarcasm. Recent approaches have addressed this issue by generating datasets for automatic sarcasm detection that reflect this discrepancy. For example, the iSarcasm dataset (Oprea and Magdy, 2020a) manually collects and labels sarcastic utterances by their authors, instead of relying on third-party annotators. However, this dataset only contains 777 sarcastic tweets and does not include perceived sarcasm. In contrast, the SPIRS dataset (Shmueli et al., 2020) utilizes reactive supervision to collect both intended and perceived sarcasm. The dataset consists of 30k tweets and relies on cues from participants in online conversations, therefore using context-aware annotations.

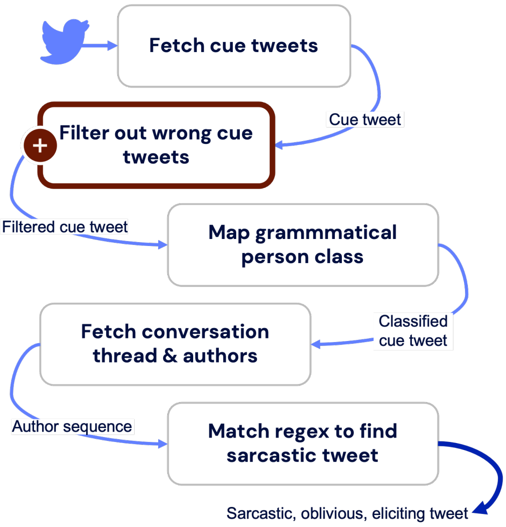
Fetch cue tweets
Cue tweet
Filter out wrong cue tweets
+
Filtered cue tweet
Map grammmatical person class
Classified cue tweet
Fetch conversation thread & authors
Author sequence
Match regex to find sarcastic tweet
Sarcastic, oblivious, eliciting tweet

Figure 1: 5-step pipeline of enhanced reactive supervision.

Models for Automatic Sarcasm Detection Various previous works emphasize the importance of contextual representations for sarcasm detection. One method uses author's behavioral trait features using different techniques (Bamman and Smith, 2015). Amir et al. 2016 proposed the usage of paragraph2vec (Le and Mikolov, 2014) over the historical utterance of users creating the user2vec model, placing similar users into nearby regions of the embedding space. On the other hand, (Zhang et al., 2016) build a deep learning model to combine text features with contextual tweets for sarcasm classification. In addition, several works have focused on different user features like behavior traits (Rajadesingan et al., 2015), user sentiment priors over entities (Khattri et al., 2015), style and personality features (Hazarika et al., 2018), or social network interactions (Plepi and Flek, 2021).

While we focus on combining different contextual text features, several studies have been dedicated to detecting sarcasm in a multimodal setting. Such works utilize information from different modalities, mainly images, and text features, and aim to capture cross-modal context for sarcasm classification (Pan et al., 2020; Xu et al., 2020; Wen et al., 2023).

Cue tweet indication	Gold	4-step	5-step
Sarcastic	318	24	109
Non-sarcastic	182	21	6
Total	500	45	115

Table 1: Comparison of the 4- and the 5-step data collection pipeline.

3 Proposed Method

3.1 Dataset Collection and Labeling

For the collection and labeling of intended and perceived sarcastic tweets, we focus on the reactive supervision method (Shmueli et al., 2020) using tweets from social media

The existing reactive supervision approach consists of four steps:

1. Fetching cue tweets q_n, querying for tweets containing "being sarcastic"

2. Mapping the cue tweets to a grammatical person class (1st, 2nd, 3rd) by examining the personal subject pronoun in the cue tweet

3. For a cue tweet q_i, fetching the corresponding conversation $C^i = \{c_n, ..., c_1\}$, where c_n is the main post, $c_1 = q_i$ and the corresponding tweet author sequence $A^i = \{a_n, a_{n-1}, ..., a_1\}$

4. Applying specific regular expressions on the author sequence to identify the sarcastic tweet. Unmatched sequences are discarded and matched are saved along with the cue tweet and the eliciting[1] and oblivious[2] tweets.

After manual analysis of random data points in the dataset (Shmueli et al., 2020), we found that the proposed approach can mistakenly label certain non-sarcastic tweets as sarcastic. We discovered several cue tweets containing "being sarcastic" which are noisy reactions from the audience, which express doubt, or ask for clarification for example: "*@user I can't tell if you are being sarcastic*".To create a dataset excluding those falsely classified tweets we propose an extension of the reactive supervision method. We add an additional filter (Figure 1), to remove tweets falsely identified as cue tweets using regular expressions, hence improving the quality of the extracted data. The filter contains a series of regular expressions to clear out the false positive cue tweets. We show a list of these regular expressions in Appendix A. In order to compare both methods, we collected 500 random cue tweets, which we labeled manually into three classes: sarcastic, non-sarcastic, and unknown (the user is asking for clarification, rather than pointing out sarcasm). Given the cue tweet and the conversation, we annotated the examples into three categories: sarcastic, non-sarcastic, and unknown. Fleiss' Kappa inter-annotator agreement between two annotators was almost a perfect agreement, with a kappa value of 0.94. Upon manual inspection and discussion, we found that the cases where the annotators were disagreeing were mainly between classes unknown and sarcastic (possible perceived sarcasm), where the user was expressing doubts if the previous tweet was sarcastic or not. Hence, we were able to resolve the disagreements through deeper inspection of the conversation thread. In Table 1 we show the number of tweets filtered out as sarcastic from both methods and also the false positive rate (we treat unknown and non-sarcastic as a single category). We observed that the number of filtered sarcastic tweets increased while, the rate of false positive examples decreased from 46.6% to 5%.

Person	Perspective	Cue tweet
1st	Intended	@user @user **I was being sarcastic**. That is what they tried to spin after the Nazi speech.
2nd	Perceived	@user I know **you are being sarcastic** btw. I just figure answering honestly is the best policy.
3rd	Perceived	@user @user Do you not see how many repeats there are? **He's being sarcastic**.

Table 2: Exemplary cue tweets per grammatical person class.

Pers.	Perspective	Sarcastic	Oblivious	Eliciting
1st	Intended	12574	12574	9023
2nd	Perceived	3295	0	519
3rd	Perceived	846	846	120
–	Non-sarc.	18535	4346	10639
Total		**35250**	**17766**	**20301**

Table 3: Break down by grammatical person class and perspective of our new dataset.

[1] Occurring if the sarcastic tweet is a reply and represents tweets which evoked the sarcastic reply (Shmueli et al., 2020)

[2] A reply to the sarcastic tweet that lacks awareness of sarcasm (Shmueli et al., 2020)

3.2 Data Statistics and Analysis

We applied our method (Figure 1) on a large scale to collect a dataset for sarcasm detection. For the collection of cue tweets, we queried for English tweets containing "being sarcastic", which are not retweets and were generated in the period from January until November 2022. For the collection of non-sarcastic tweets, we chose to fetch tweets randomly, querying for English tweets that have been generated from January until November 2022, are not retweets, and don't contain the words "sarcastic", "sarcasm" or the tags "#sarcasticquote", "#sarcasticquotes", "#sarcasticmemes", "#sarcastic", "#sarcasm". Finally, we gathered 17k English sarcastic tweets and 19k non-sarcastic tweets with corresponding additional conversational contexts such as oblivious or elicit tweets (a tweet that caused the sarcastic reply). In addition, we collected around 89M historical tweets for the users in our dataset in order to extend the dataset with additional author contextual information.

Statistics We collected 100K cue tweets for the new dataset. In Table 2 we present examples of the cue tweet for each grammatical person class. Next, we applied the exclusive filter, filtering out 26.6% of the cue tweets. After collecting the threads, and corresponding authors for the remaining cue tweets and matching those author sequences, we end up with 17k English sarcastic tweets, 10k eliciting, and 13k oblivious tweets. In addition, we collected 19k non-sarcastic tweets as well as 11k corresponding eliciting and 4k oblivious tweets. We summarize the new dataset grouped by grammatical person classes and perspectives in Table 3, and with the statistics of user history in Table 4.

In Table 5 we examine the distribution of different author sequence patterns of the sarcastic threads. We observed that 80% of the threads are equal to or smaller than 4 tweets per thread. In addition, it shows the most common author thread pattern per grammatical-person class, indicating that sarcastic tweets are often provoked by other authors (see eliciting tweets). Moreover, we notice the patterns used to detect perceived sarcasm, grouped in 2nd and 3rd person perspective cues. These cues capture conversations where other participants detect the presence of sarcasm.

During our analysis of the most common bi-grams in the dataset, we noticed that political or politician-related bi-grams predominated within the perceived sarcasm class (Figure 2). This finding reinforces the link between sarcasm and political discourse (Davis et al., 2018; Khodak et al., 2017), offering insights into the potential significance of the detection of (perceived) sarcasm in understanding the political stance and the presence of this linguistic phenomenon in online interactions.

Class/Perspective	# Authors	# Historical tweets
Sarcastic	**15884**	**45244265**
Intended	12245	33328130
Perceived	3686	12257193
Both	47	–
Non-sarcastic	**17340**	**43475563**
Both	**99**	–
Total	**33125**	88719828

Table 4: Break down of the number of tweet authors by class and perspective.

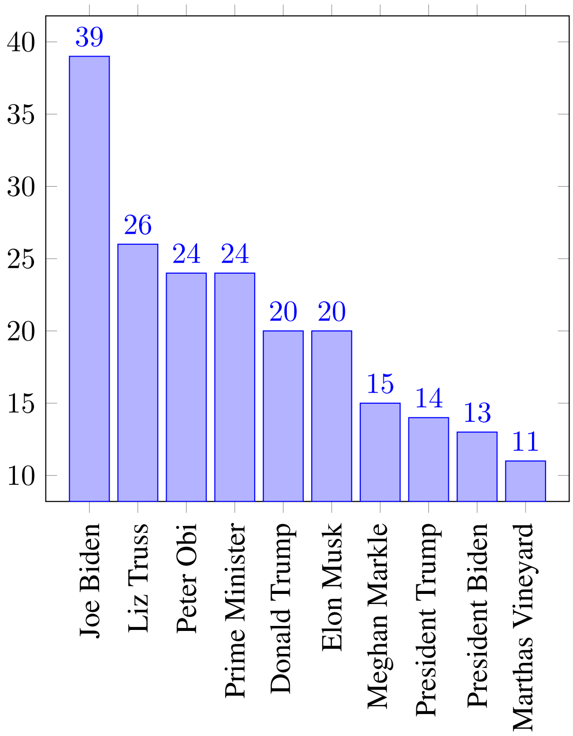

Figure 2: Top 10 most common bi-grams in as sarcastic perceived tweets.

Historical tweets The 35k sarcastic and non-sarcastic tweets of our new dataset have been composed by 33k different authors. Along with the new dataset we collected 89M historical tweets for those 32k authors (Table 4). The number of historical tweets per author varies between 1 (16 authors have 1 historical tweet) and 500 (upper bound) with an average tweet number of 471.46.

Person	Pattern	Count	% of person class
1st (intended)	*ABAC*	3368	27%
	ABA	2795	22%
	ABAB	1918	15%
	other	4493	36%
Subtotal		12574	
2nd (perceived)	*AB*	2679	82%
	ABA	476	14%
	other	140	4%
Subtotal		3295	
3rd (perceived)	*ABC*	621	73%
	ABCA	54	6%
	other	171	20%
Subtotal		846	
Total		16715	

Table 5: Most common thread pattern by person class. The colors represent cue, oblivious, sarcastic and eliciting tweets. The shown letters correspond to different authors in the thread. Equal letters encode equal authors, and the author sequences are shown in reverse order. The rightmost letter represents the end of the thread (cue tweet) while the leftmost represents the beginning of the thread.

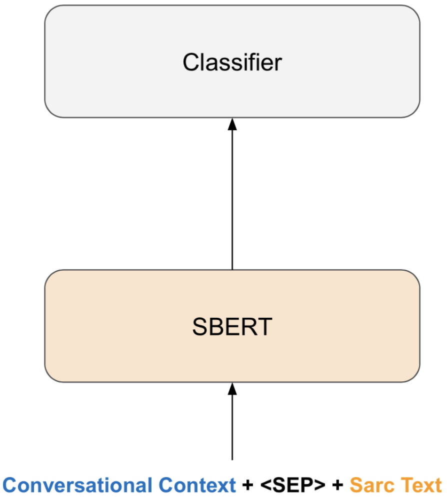

Figure 3: For the conversational context, we still use SBERT model as our base model. We only append the conversational context (namely oblivious and elicit tweet) to the original tweet to be classified and separated with special tokens.

4 Methodology

The models used for our experiments can be divided into two model groups: Text-only-based models and author-contextual-based models.

4.1 Text-only-based models

This model only uses a representation of the textual information in the sarcastic and non-sarcastic tweets as input. For this purpose, we fine-tuned the pre-trained Transformer encoder like SentenceBERT (Reimers and Gurevych, 2019) on the binary task of predicting the label sarcastic vs. non-sarcastic or perceived vs. intended, given only the tweet text. In this setup, we are also able to append the conversational context, namely oblivious and elicit tweet [3], in case those exist. We do so by appending the conversational context with the tweet that is to be classified, and we use special tokens to separate those (as in Figure 3).

4.2 Author Contextual Models

These models expand the textual features of tweets by adding representations of the authors of tweets as features. For encoding user representations, we used different models similar to Plepi et al. (2022a), namely: a) Priming, b) Average SentenceBERT for authors (A-SBERT), c) Authorship Attribution (AA) d) Graph Neural Networks (GNN).

Priming For our purpose, we randomly sample a number of tokens and append them as a prefix to the tweet text to classify. For each author a, we randomly sample a number of tokens from their historical tweets H^a (consisting of a sequence of historical tweets $\{h_1, h_2, ..., h_n\}$ and $|w_i|$ corresponding to the number of tokens/words in the tweets) until the maximum number of tokens is less than 200 or corresponds to the number of tokens in their historical tweets $\sum_{i=1}^{n} |w_i|$, if $\sum_{i=1}^{n} |w_i| < 200$. We append the sampled text to the beginning of the tweet text, which is to be classified during fine-tuning of SentenceBERT.

Average SentenceBERT for authors (A-SBERT) Given an author a and their historical tweets, H^a. We compute the author representation by averaging the SentenceBERT tweet embeddings h_i' of all $h_i \in H^a$, resulting in: $\bar{a} = \frac{1}{|H^a|} \sum_{k=1}^{|H^a|} h_i'$.

Authorship Attribution (AA) With this technique, we pre-train a neural network to predict the author of a given tweet, $p(a|t_i')$. We forward the SentenceBERT tweet embeddings t_i' into a two-layer feed-forward network parameterized from weight matrices $W_1 \in R^{\frac{d}{2} \times d}$ and $W_2 \in R^{n \times \frac{d}{2}}$, where d is 768 (dimension of the SentenceBERT tweet embeddings), and $n \equiv$ number of authors during the training. Then, we forward the output of the last linear layer to a softmax layer to get a

[3] Cue tweets are not part of the conversational context.

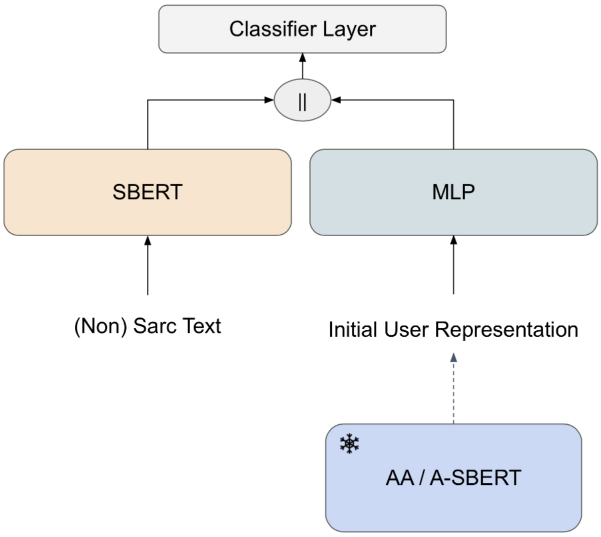

Figure 4: In this figure we show how we combine precomputed user representation with SBERT. A-SBERT, and AA, are separate encoding methods, to extract initial user representations, utilizing their comments during the history. After computing those, we combine both user and text representations to classify. The encoding layer is frozen during training.

distribution over the authors. After training, we use the linear layers to extract a representation of the author. For each author a, we forward all their historical tweets H^a to the trained network, extracting the predictions, $Y = \{y_h | h \in H^a\}$. Next, we initialize a vector of size n, where $\bar{a}_i =| \{y | y \in Y \wedge y = i\} |$, for $i = (1, ..., n)$, representing the number of times each author is predicted for all tweets of a. We extend this representation by normalizing the vector, so that the sum of all predictions is equal to 1 and thus get another representation - the distribution of authors predicted.

A-SBERT, and AA, are separate encoding methods to extract initial user representations, utilizing their comments during the history. After computing those, we combine both user and text representations, as in Figure 4 to classify the text.

Graph Neural Network (GNN) In this model, we aim to model the social relations between users, and the relations between tweets and users. For this purpose, we build a heterogeneous graph $\mathcal{G} = (V, E)$, where $V = \{U \cup T\}$, which consists of two types of nodes: users and tweets (Plepi and Flek, 2021). In order to model both types of relations, we use two types of edges $E = \{\mathrm{e}^U \cup \mathrm{e}^T\}$, where e^U represents the social interaction between users [4], and e^T represents the relation between an author and his tweet. Finally, we use Graph Attention Networks (GATs, (Veličković et al., 2018)) to learn the representations of the nodes in the graph. In recent works, GNNs have shown improvements in the performance for various NLP tasks (Mishra et al., 2019a,b; Kacupaj et al., 2021; Sakketou et al., 2022; Plepi et al., 2022b).

We then combine the SentenceBERT model, fine-tuned on the binary task of predicting the label (sarcastic vs. non-sarcastic and intended vs. perceived), given the tweet with an additional layer concatenating the tweet with the author representation computed using Average SentenceBERT for authors or Authorship Attribution. For priming, we also use the SentenceBERT model but fine-tuned to the binary task of predicting the label (sarcastic vs. non-sarcastic and intended vs. perceived), given the sampled text from each author and the sarcastic/non-sarcastic tweet.

5 Experimental Setup

Our experiments are focused on two main tasks: sarcasm detection to predict if a tweet is sarcastic or not, and perspective classification to predict if a sarcastic tweet is intended or perceived. We utilized our new dataset, consisting of 35K tweets, to train our text-only-based models. On the other hand, to train the author-contextual-based models, we also included historical tweets to precompute user representations.

5.1 Implementation details

We split both datasets randomly along the tweet IDs. Splitting them into 80% training and 20% testing tweets. For all models, we use a dropout of 0.2, the Adam optimizer with a learning rate of $1e-4$ and weighted cross entropy loss. Each model was trained for a total of 10 epochs, with a batch size of 32, and was saved each time the performance on the validation set is topped. We pre-processed the data using the DistilRoBERTa (Sanh et al., 2019) Tokenizer[5]. We replaced mentions of users with $@user$, encoded emojies with text, removed URLs, non-ASCII characters and digits. The dataset and the code repository for reproducibility are available

[4] Interactions on Twitter include quoting, mentioning, or replying

[5] https://huggingface.co/sentence-transformers/all-distilroberta-v1

Dataset	Model	F1	Accuracy
$N = 34938$	SBERT	74.4	74.5
	Priming	77.5	77.7
	AA	79.3	79.3
	A-SBERT	80.1	80.1
	GNN	**82.0**	**82.2**

Table 6: Accuracy and macro F1 scores as percentages for sarcasm detection.

here https://github.com/caisa-lab/konvens2023-sarcasm-detection.git.

6 Results and Analysis

6.1 Sarcasm Detection

Our initial experiments focused on the task of sarcasm detection, and we show the results in Table 6. As also seen in previous works (Bamman and Smith, 2015; Amir et al., 2016; Plepi and Flek, 2021), author-contextual-based models outperform text-based models. The additional context from the author's representations enriches the text features and enhances its performance on the task of sarcasm detection.

Our results' analysis revealed that GNN based model is our best-performing one with an 82.2% F1-score. Modeling social network interactions as graphs proves to be an effective way to learn better representations for both text and users. Furthermore, author attribution performed slightly worse than A-SBERT, mainly due to sparsity in AA representation. Another limitation of AA is its scaling over more authors. Overall, GNN and A-SBERT proved to be the most effective in terms of both performance and computational costs, due to no additional training for computing the author representation.

6.2 Conversational context

In addition, we also incorporate conversational context, which includes oblivious and eliciting tweets into our models. [6] We observe an improvement in all our models, where the most significant one is for the text-only SBERT model, with 10.4%. Interestingly, the model that gains less from the conversational context is the GNN model with only 1.3% (Table 7). One reason for this might be due to the way in which the GNN model incorporates the additional context. In the GNN model, the oblivious and eliciting tweets are added as separate nodes in the graph, while for the other models, we incorporate the conversational context by concatenating with the text to be classified. The best-performing model in this setup is the author attribution-based model.

Dataset	Model	F1	Accuracy
$N = 34938$	o/e SBERT	84.9	84.9
	o/e A-SBERT	85.0	85.0
	o/e AA	**85.6**	**85.5**
	o/e GNN	83.0	83.5

Table 7: Accuracy and macro F1 scores as percentages for sarcasm detection. O/e indicates the usage of eliciting and oblivious tweets.

Dataset	Model	F1	Accuracy
$N = 16278$	SentenceBERT	68.5	79.2
	Priming	70.9	79.8
	A-SBERT	70.6	79.2
	AA	71.3	**82.2**
	GNN	**72.2**	80.8

Table 8: Accuracy and macro F1 scores as percentages for perspective classification.

6.3 Sarcasm Perspective Classification

Finally, we also experimented with the perspective classification task. Here, we face an imbalanced dataset, where 75.2% is intended sarcasm and 24.8% is perceived sarcasm. Our results for this task are shown in Table 8. We notice a lower improvement of at most only 3.0%, of author-contextual-based models over the SBERT model compared to sarcasm detection task, where the improvement was up to 7.6%. These results also align with the conclusion in (Oprea and Magdy, 2019; Plepi and Flek, 2021), on the perception classification task. Hence, we believe that modelling the representation of the author is less useful for the classification of perceived sarcasm. To increase the number of tweets classified as perceived, it could be of benefit to additionally model user embeddings for the audience of the tweet, predicting how individual users will react towards the tweet.

6.4 Error Analysis

Generally, we found that in the perception classification task, perceived tweets are harder to detect than intended sarcasm, which is in line with the results of (Oprea and Magdy, 2019; Plepi and Flek, 2021). This challenge is caused not only by the imbalance but also by the complexity of perceived sarcasm, and how the text is interpreted from the broad audience on Twitter. Table 9 presents the percent-

[6] Except priming due to the maximum length limitation that can be taken as an input to the SBERT model.

Model	$\mathbf{F_I}$	$\mathbf{F_P}$
SBERT	59.1	7.5
Priming	50.9	9.9
A-SBERT	49.2	11.4
AA	58.5	4.5
GNN	51.4	7.8
o/e SBERT	50.4	7.8
o/e A-SBERT	39.9	9.1
o/e AA	38.3	10.6
o/e GNN	50.6	8.2

Table 9: False predicted sarcastic perspectives as percentages in relation to gold labels for all models used. F_I is the percentage of perceived tweets falsely classified as intended; F_P, the percentage of intended tweets falsely classified as perceived. Number of test instances: 3343 tweets.

ages of misclassified examples for both perceived and intended sarcasm across different models. In the first part, we show the models without conversational context. Consistently across all models, one can observe a higher percentage of misclassified perceived sarcasm compared to intended sarcasm. Improving the quality and quantity of perceived sarcasm remains a challenging task, given its subjective nature that is often influenced by the audience's diverse social and cultural backgrounds, which may influence their interpretation of tweets on a certain topic. However, as the performance improves by adding the conversational context, it seems that the improvement comes mainly from the classifications of the perceived tweets. We notice a significant drop in the percentage for false classified perceived tweets as intended. These results show the importance of exploring the use of additional context that involves the audience to enhance the detection of perceived sarcasm.

7 Conclusions

In this work, we present an improvement of reactive supervision, in order to collect higher-quality data for the sarcasm detection task. Our manual analysis indicates a reduced number of false positives due to the reduction of noise in the sarcastic data, and removal of unclear cues. In addition, we also collect conversational and author context for our dataset in order to enhance the performance in the sarcasm detection task. Our findings show the importance of additional context in both the sarcasm detection task and the perception classification.

Limitations

Our dataset was collected only in the English language, and the dataset might be focused more on English speakers' sarcasm. In addition, the amount of perceived sarcasm that we collected is lower than the intended sarcasm. The main reason is the complexity of the perspective sarcasm, and the difficulty in solving cases that request additional clarification from the users. Future work can focus more on analyzing these cases by taking into account the topic where the potential sarcastic comment was made and also the communities in social media that may perceive such text as sarcastic. Moreover, it might be interesting to include an additional sarcastic type that is both intended and perceived. However, this type might be difficult to capture using distant supervision, and might need to be combined with additional manual annotation of the conversational thread where the cue tweet is happening. In our experiments, we used a pretrained model SBERT (Reimers and Gurevych, 2019); however, the results might slightly differ with the usage of bigger and more recent pretrained models. Finally, we did not focus on extracting different demographic features from the historical data of the users. Such features might improve the analysis and understanding of the perceived sarcasm (Oprea and Magdy, 2020b). In addition, one could explore adding feature with respect to the political topics, such as political bias in a conversation, in order to improve conversational features for the sarcasm detection task (Kannangara, 2018; Ghosh et al., 2020).

Ethical Considerations

Improving the performance of artificial agents by modeling the personal characteristics of online users' language requires careful consideration of a wide range of ethical concerns.

To ensure data privacy, all collected user history is kept separately on protected servers, linked to the raw text only through hashed anonymous IDs for each user. The collected dataset is solely limited to the purpose of this study for sarcasm detection, and no individual posts shall be republished (Hewson and Buchanan, 2013). Moreover, we utilize publicly available Twitter data in a purely observational (Norval and Henderson, 2017) and non-intrusive manner.

The use of models that incorporate contextual user information may carry the risk of invoking stereotyping and essentialism, as the models may lean toward labeling people rather than posts (Rudman and Glick, 2008). Therefore, it is crucial to remain mindful of these effects when interpreting the model results in its own end-application context.

References

Brian P. Ackerman. 1982. Contextual integration and utterance interpretation: The ability of children and adults to interpret sarcastic. *Child Development, Volume 53*, pages 1075–1083.

Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 167–177, Berlin, Germany. Association for Computational Linguistics.

David Bamman and Noah Smith. 2015. Contextualized sarcasm detection on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 9.

Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. 2014. Modelling sarcasm in Twitter, a novel approach. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 50–58, Baltimore, Maryland. Association for Computational Linguistics.

Alexandru-Costin Băroiu and Stefan Trăusan-Matu. 2022. Automatic sarcasm detection: Systematic literature review. *Information* ***2022****, 13, 399.*

Paula Carvalho, Luís Sarmento, Mário J Silva, and Eugénio De Oliveira. 2009. Clues for detecting irony in user-generated contents: oh...!! it's" so easy";-. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 53–56.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010a. Semi-supervised recognition of sarcasm in twitter and amazon. In *Proceedings of the fourteenth conference on computational natural language learning*, pages 107–116.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010b. Semi-supervised recognition of sarcasm in Twitter and Amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 107–116, Uppsala, Sweden. Association for Computational Linguistics.

Jenny L Davis, Tony P Love, and Gemma Killen. 2018. Seriously funny: The political work of humor on social media. *New Media & Society*, 20(10):3898–3916.

Shelly Dews, Joan Kaplan, and Ellen Winner. 1995. Why not say it directly? the social functions of irony. *Discourse processes*, 19(3):347–367.

Megan L. Dress, Roger J. Kreuz, Kristen E. Link, and Gina M. Caucci. 2008. Regional variation in the use of sarcasm. *Journal of Language and Social Psychology*, 27(1):71–85.

Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 392–398, Istanbul, Turkey. European Language Resources Association (ELRA).

Debanjan Ghosh, Avijit Vajpayee, and Smaranda Muresan. 2020. A report on the 2020 sarcasm detection shared task. *arXiv preprint arXiv:2005.05814*.

Raymond W Gibbs. 1986. On the psycholinguistics of sarcasm. *Journal of experimental psychology: general*, 115(1):3.

Roberto González-Ibánez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586.

Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. 2018. Cascade: Contextual sarcasm detection in online discussion forums. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1837–1848.

Claire Hewson and Tom Buchanan. 2013. Ethics guidelines for internet-mediated research. The British Psychological Society.

Aditya Joshi, Pushpak Bhattacharyya, and Mark James Carman. 2016. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50:1 – 22.

Endri Kacupaj, Joan Plepi, Kuldeep Singh, Harsh Thakkar, Jens Lehmann, and Maria Maleshkova. 2021. Conversational question answering over knowledge graphs with transformer and graph attention networks. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 850–862, Online. Association for Computational Linguistics.

Sandeepa Kannangara. 2018. Mining twitter for fine-grained political opinion polarity classification, ideology detection and sarcasm detection. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 751–752.

Anupam Khattri, Aditya Joshi, Pushpak Bhattacharyya, and Mark Carman. 2015. Your sentiment precedes you: Using an author's historical tweets to predict sarcasm. In *Proceedings of the 6th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 25–30.

Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. 2017. A large self-annotated corpus for sarcasm. *arXiv preprint arXiv:1704.05579*.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, page II–1188–II–1196. JMLR.org.

Pushkar Mishra, Marco Del Tredici, Helen Yannakoudakis, and Ekaterina Shutova. 2019a. Abusive Language Detection with Graph Convolutional Networks. 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)*, Minneapolis, Minnesota.

Rohan Mishra, Pradyumn Prakhar Sinha, Ramit Sawhney, Debanjan Mahata, Puneet Mathur, and Rajiv Ratn Shah. 2019b. SNAP-BATNET: Cascading author profiling and social network graphs for suicide ideation detection on social media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 147–156, Minneapolis, Minnesota. Association for Computational Linguistics.

Chris Norval and Tristan Henderson. 2017. Contextual consent: Ethical mining of social media for health research. *CoRR*, abs/1701.07765.

Silviu Oprea and Walid Magdy. 2019. Exploring author context for detecting intended vs perceived sarcasm. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2859, Florence, Italy. Association for Computational Linguistics.

Silviu Oprea and Walid Magdy. 2020a. 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.

Silviu Vlad Oprea and Walid Magdy. 2020b. The effect of sociocultural variables on sarcasm communication online. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW1):1–22.

Hongliang Pan, Zheng Lin, Peng Fu, Yatao Qi, and Weiping Wang. 2020. Modeling intra and inter-modality incongruity for multi-modal sarcasm detection. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1383–1392, Online. Association for Computational Linguistics.

Joan Plepi and Lucie Flek. 2021. Perceived and intended sarcasm detection with graph attention networks. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4746–4753, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Joan Plepi, Béla Neuendorf, and Lucie Flek. 2022a. Unifying data perspectivism and personalization: An application to social norms. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7391–7402. Association for Computational Linguistics.

Joan Plepi, Flora Sakketou, Henri-Jacques Geiss, and Lucie Flek. 2022b. Temporal graph analysis of misinformation spreaders in social media. In *Proceedings of TextGraphs-16: Graph-based Methods for Natural Language Processing*, pages 89–104, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Tomás Ptácek, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on czech and english twitter. In *COLING 2014*, pages 213–223.

Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 2015. Sarcasm detection on twitter: A behavioral modeling approach. In *Proceedings of the eighth ACM international conference on web search and data mining*, pages 97–106.

Nils Reimers and Iryna Gurevych. 2019. entence- bert: Sentence embeddings using siamese bert- networks. In *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992. Association for Computational Linguistics.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in twitter. *Language resources and evaluation*, 47(1):239–268.

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.

Patricia Rockwell and Evelyn M. Theriot. 2001. Culture, gender, and gender mix in encoders of sarcasm: A self-assessment analysis. *Communication Research Reports*, 18(1):44–52.

Laurie A Rudman and Peter Glick. 2008. The social psychology of gender: How power and intimacy shape gender relations.

Flora Sakketou, Joan Plepi, Riccardo Cervero, Henri Jacques Geiss, Paolo Rosso, and Lucie Flek. 2022. FACTOID: A new dataset for identifying misinformation spreaders and political bias. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3231–3241, Marseille, France. European Language Resources Association.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Boaz Shmueli, Lun-Wei Ku, and Soumya Ray. 2020. Reactive supervision: A new method for collecting sarcasm data. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 2553–2559.

Oren Tsur, Dmitry Davidov, and Ari Rappoport. 2010. Icwsm—a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 4.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations*. Accepted as poster.

Byron C. Wallace, Do Kook Choe, Laura Kertz, and Eugene Charniak. 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, Baltimore, Maryland. Association for Computational Linguistics.

Changsong Wen, Guoli Jia, and Jufeng Yang. 2023. Dip: Dual incongruity perceiving network for sarcasm detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2540–2550.

Nan Xu, Zhixiong Zeng, and Wenji Mao. 2020. Reasoning with multimodal sarcastic tweets via modeling cross-modality contrast and semantic association. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3777–3786, Online. Association for Computational Linguistics.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. Tweet sarcasm detection using deep neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: technical papers*, pages 2449–2460.

Regular expressions
`r"not being sarcastic"`
`r"not(\s*[A-Za-z,;’\\\/s@])*\s*sarcastic") r"(sarcastic)\s*(\?)+"`
`r"wasn’t being sarcastic"`
`r"wasnt being sarcastic"`
`r"wasn’t being sarcastic"`
`r"was not being sarcastic"`
`r"weren’t being sarcastic"`
`r"weren’t being sarcastic"`
`r"werent being sarcastic"`
`r"were not being sarcastic"`
`r"(sarcastic)\s*(\?)+"`
`r"sarcastic\sor"`
`r"hope(\s*[A-Za-z,;’\\s@])*\s*being sarcastic"`
`r"hope(\s*[A-Za-z,;’\\s@])*\s*being(\s*A-Za-z,;’\\s@)*\s*sarcastic"`
`r"hope you’re being sarcastic"`
`r"pray(\s*[A-Za-z,;’\\s@])*\s*being sarcastic"`
`r"if(\s*[A-Za-z,;’\\s@])*\s*being sarcastic"`
`r"sarcastic[A-Za-z,;’\\s@]*\s*correct"`
`r"sarcastic\s*([A-Za-z,;’\\s@]\s)0,2 right"`
`r"are you being sarcastic"`

Table 10: Compound regular expression used to filter tweets incorrectly identified as cue tweets.

A Regular Expressions

Table 10, shows a list of curated regular expressions that we used to filter out false positive cue tweets. The main target class that was fixed from the regular expressions, was the perceived sarcasm, where the number of false positive rate was significantly reduced.