"When Words Fail, Emojis Prevail": Generating Sarcastic Utterances with Emoji Using Valence Reversal and Semantic Incongruity

Faria Binte Kader*, **Nafisa Hossain Nujat***, **Tasmia Binte Sogir***,
Mohsinul Kabir, Hasan Mahmud, Kamrul Hasan
Department of Computer Science and Engineering
Islamic University of Technology
Dhaka, Bangladesh
{faria, nafisa13, tasmia, mohsinulkabir, hasan, hasank}@iut-dhaka.edu

Abstract

Sarcasm is a form of figurative language that serves as a humorous tool for mockery and ridicule. We present a novel architecture for sarcasm generation with emoji from a non-sarcastic input sentence in English. We divide the generation task into two sub tasks: one for generating textual sarcasm and another for collecting emojis associated with those sarcastic sentences. Two key elements of sarcasm are incorporated into the textual sarcasm generation task: valence reversal and semantic incongruity with context, where the context may involve shared commonsense or general knowledge between the speaker and their audience. The majority of existing sarcasm generation works have focused on this textual form. However, in the real world, when written texts fall short of effectively capturing the emotional cues of spoken and face-to-face communication, people often opt for emojis to accurately express their emotions. Due to the wide range of applications of emojis, incorporating appropriate emojis to generate textual sarcastic sentences helps advance sarcasm generation. We conclude our study by evaluating the generated sarcastic sentences using human judgement. All the codes and data used in this study has been made publicly available[1].

1 Introduction

Sarcasm is defined as the use of remarks that often mean the opposite of what is said in order to hurt someone's feelings or to criticize something in a humorous way[2]. Sarcastic remarks are often challenging to interpret considering their literal meaning differs greatly from the speaker's actual intent. Compared to verbal or in-person conversations, textual sarcasm presents additional challenges due to the absence of visual cues, vocal tone etc.

Non-Sarcastic Input	Sarcastic Output with Emoji
I really hate walking in the rain.	I really love the outdoors walking in the rain. I sat feeling thoroughly miserable.😫
Mom is in a bad mood today.	Happy mothers day mom is in a well mood today. She sounded tense and angry.😠
That movie was bad.	That movie was awesome. Bad intelligence and political incompetence.😠

Table 1: Sample sarcastic outputs with emoji generated from non-sarcastic inputs

The presence of sarcasm makes it significantly harder for machines to understand the actual meaning of the textual data. This has motivated research in detecting sarcasm in textual data. In order to train machines to detect sarcasm, we need quality datasets that represent different aspects of sarcasm in text. Even though we have an abundance of social media data and resources, it can be difficult to collect correctly labeled sarcastic texts. Instead, many research have tried to generate texts that can accurately express sarcastic notions (Joshi et al., 2015; Mishra et al., 2019; Chakrabarty et al., 2020). Many studies have also investigated strategies in incorporating sarcasm generation into chatbots (Joshi et al., 2015, 2017).

Emojis, small ideograms that represent objects, people, and scenes (Cappallo et al., 2015), are one of the key elements of a novel form of communication due to the advent of social media. Using emojis within texts can give us additional cues on sarcasm, replicating facial expressions and body language, etc. Incorporating emojis with texts for training will let the machines catch these cues easily (Bharti et al., 2016). Subramanian et al. (2019)

*These authors contributed equally to this work.
[1] https://github.com/WrightlyRong/Sarcasm-Generation-with-Emoji
[2] https://dictionary.cambridge.org/

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics - Student Research Workshop, pages 334–351
July 10-12, 2023

observed that when emojis were included in the sentence, their emoji-based sarcasm detection model performed noticeably better.

In this study, we propose a new framework in which when given a non-sarcastic text as input, the text is converted into a sarcastic one with emoji where the emoji will specifically help to identify the sarcastic intent of the text. Table 1 shows a few sample non-sarcastic input and sarcastic output pairs with emoji. In order to implement the architecture, we have focused on two major components: Sarcastic text generation and Emoji prediction for the text. For textual sarcasm generation, we are incorporating the works of Chakrabarty et al. (2020) and Mishra et al. (2019) and for Emoji prediction, a deep learning model fine tuned on OpenAI's CLIP (Contrastive Language-Image Pre-training)[3] (Radford et al., 2021) is used. The emoji prediction module along with the sarcasm generation module generates the final sarcastic text including emoji. This work provides two major contributions:

1. Propose a novel multi-modular framework for sarcasm generation incorporating the reversal of valence and semantic incongruity characteristics of sarcasm while also including appropriate emojis.

2. Create and publish a sarcastic corpora which can serve as valuable training data for sarcasm detection models.

As far as our understanding goes, there has been no previous framework proposed on textual sarcasm generation that also incorporates emojis. This framework can aid downstream tasks by allowing a deeper understanding of sarcasm to produce more contextually relevant responses.

2 Related Work

Research on sarcasm have been a subject of interest for several decades. The following sub sections provide a brief overview of the past work done on different aspects of sarcasm.

2.1 Studies on Sarcasm Detection

Sarcasm detection is a classification task in its most typical form. From a given text, the task includes classifying the text as sarcastic or non-sarcastic. Sarcasm detection is a fairly recent but promising research field in the domain of Natural Language Processing. Nonetheless, it serves as a crucial part to sentiment analysis (Maynard and Greenwood, 2014).

Most of these studies on sarcasm detection train and test on already available popular datasets such as the datasets used by Riloff et al. (2013), Khodak et al. (2017) and Cai et al. (2019). We observed that Twitter is predominantly the most popular social media platform used for sarcasm detection datasets although Reddit, Amazon and a few discussion forums were also seen being used. We also saw a shift in Sarcasm detection methodologies from rule-based approaches (Riloff et al., 2013; Bharti et al., 2015), machine learning and deep learning approaches (Bharti et al., 2017; Poria et al., 2016; Ghosh and Veale, 2016) to transformed based approaches (Dadu and Pant, 2020; Kumar et al., 2021). We include two tables Table 9 and Table 10 summarizing the datasets and methodologies used in sarcasm detection in the appendix (Section A).

Recent works on sarcasm detection include frequent use of BERT (Savini and Caragea, 2022; Zhang et al., 2023; Pandey and Singh, 2023), multimodal and cross-modal detection tasks (Liang et al., 2022; Chauhan et al., 2022; Ding et al., 2022), enhancement of sarcasm detection in complex expressions with sememe knowledge (Wen et al., 2022), study on the effect of foreign accent (Puhacheuskaya and Järvikivi, 2022), use of vocal and facial cues (Aguert, 2022) etc. Sarcasm and irony detection from languages other than English i.e. Chinese, Dutch, Spanish, Arabic, Romanian etc. have also been studied in recent works (Farha and Magdy, 2020; Muaad et al., 2022; Maladry et al., 2022; Wen et al., 2022; Ortega-Bueno et al., 2022; Buzea et al., 2022).

2.2 Characteristics of Sarcasm

Studies have identified a variety of potential sources for sarcasm. According to Gerrig and Goldvarg (2000), sarcasm stems from a situational disparity between what the speaker desires, believes, or expects and what actually happens. Incongruity between text and a contextual information is mentioned as a factor by Wilson (2006). Context Incongruity (Campbell and Katz, 2012) is addressed in the works of Riloff et al. (2013) who suggests that sarcasm arises from a contrast between positive verbs and negative situation phrases. Burgers et al. (2012) formulates that for an utterance to be

[3] https://openai.com/research/clip

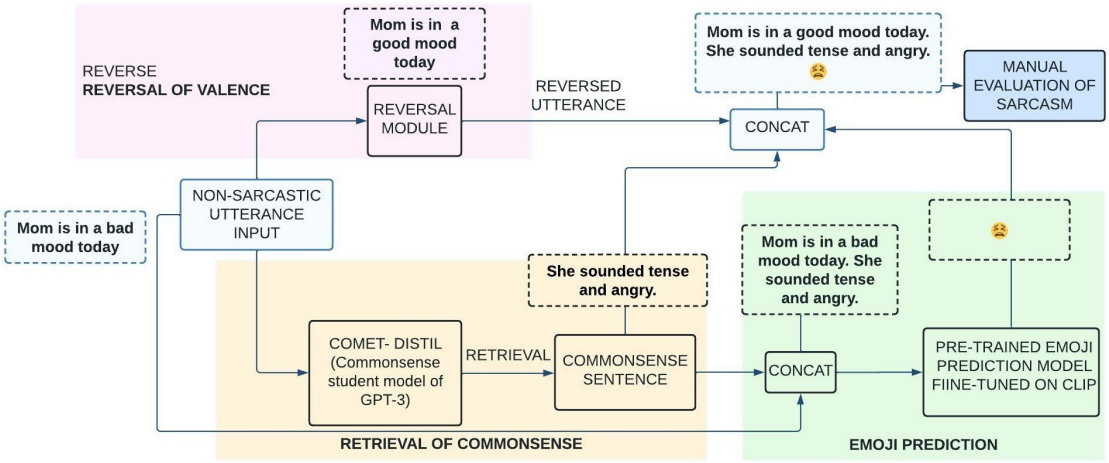

Figure 1: Model Architecture of the proposed system

sarcastic, it needs to have one or more of these five characteristics:

1. the sentence has to be evaluative,
2. it should be based on the reversal of valence of the literal and intended meanings,
3. it should have a semantic incongruity with the context, which may consist of common sense or general information that the speaker and the addressee share,
4. should be aimed at some target,
5. should be in some manner relevant to the communication scenario. Many studies focused on one or more of these characteristics.

2.3 Sarcasm Generation

Compared to sarcasm detection, research on sarcasm generation is still in its early stages. Joshi et al. (2015) introduced SarcasmBot[4], a chatbot that caters to user input with sarcastic responses. SarcasmBot is a sarcasm generation module with eight rule-based sarcasm generators where each of the generators produces a different type of sarcastic expression. During the execution phase, one of these generators is selected based on user input properties. Essentially, it yields sarcastic responses rather than converting a literal input text into a sarcastic one, the latter one being a common practice in future research. This method was later utilized in the author's subsequent work (Joshi et al., 2017) where they built SarcasmSuite, a web-based interface for sarcasm detection and generation.

The first work on automatic sarcasm generation conditioned from literal input was performed by Mishra et al. (2019). The authors relied on the Context Incongruity characteristic of sarcasm mentioned by Riloff et al. (2013) and employed information retrieval-based techniques and reinforced neural seq2seq learning to generate sarcasm. They used unlabeled non-sarcastic and sarcastic opinions to train their models, where sarcasm was formed as a result of a disparity between a situation's positive sentiment context and negative situational context. A thorough evaluation of the proposed system's performance against popular unsupervised statistical, neural, and style transfer techniques showed that it significantly outperformed the baselines taken into account.

Chakrabarty et al. (2020) introduced a new framework by incorporating context in the forms of shared commonsense or world knowledge to model semantic incongruity. They based their research on the factors addressed by Burgers et al. (2012). Their architecture is structured into three modules: Reversal of Valence, Retrieval of Commonsense Context, and Ranking of Semantic Incongruity. With this framework they were able to simulate two fundamental features of sarcasm: reversal of valence and semantic incongruity with the context. However, they opted for a rule-based system to reverse the sentiments. The authors also noticed that in a few cases, the simple reversal of valence strategy was enough to generate sarcasm which meant the addition of context was redundant.

Recent similar works in the field include that of Oprea et al. (2021) where they developed a sarcastic response generator, Chandler, that also provides explanations as to why they are sarcastic. Das et al. (2022) manually extracted the features of a

[4] https://github.com/adityajo/sarcasmbot/

benchmark pop culture sarcasm corpus and built padding sequences from the vector representations' matrices. They proposed a hybrid of four Parallel LSTM Networks, each with its own activation classifier which achieved 98.31% accuracy among the test cases on open-source English literature. A new problem of cross-modal sarcasm generation (CMSG) that creates sarcastic descriptions of a given image was introduced by Ruan et al. (2022). However, these studies have only focused on generating textual sarcastic sentences, but as described by Subramanian et al. (2019), incorporating emojis improved the overall performance of sarcasm detection and thus can be a potential research scope.

3 Methodology

Our model architecture consists of 3 modules which are as follows: Reversal of Valence, Retrieval of Commonsense and Emoji Prediction. The Reversal of Valence module takes in a negative utterance and generates an utterance with positive sentiment. The Retrieval of Commonsense module outputs relevant commonsense context sentence which helps in creating a sarcastic situation. Lastly, the Emoji Prediction module generates an emoji which makes the overall output more sarcastic. With these three modules, we have incorporated two of the fundamental features of sarcasm: reversal of valence and semantic incongruity with the context. A diagram of the overall pipeline is demonstrated in Figure 1. We describe the modules in details in the next few sub sections.

3.1 Reversal of Valence

In the work of Chakrabarty et al. (2020), for the reversal of valence module, they have used a rule-based approach to manually reverse the sentiment of the negative sentence. But a rule-based model cannot reverse sentences that do not follow the traditional structure of sentences such as those used in social media. We have worked on this limitation of this current state-of-the-art sarcasm generation model where we replace their rule-based reversal module with a deep-learning reversal module inspired by the work of Mishra et al. (2019). This module is divided into two parts: Sentiment Neutralization and Positive Sentiment Induction.

3.1.1 Sentiment Neutralization

We implement the Sentiment Neutralization module to filter out the sentiment words from the input utterance, which results into a neutral sentence from a negative one. An example is shown in table 2.

Negative Input	Neutral Output
Is feeling absolutely bloated and fat from lack of a proper workout	Is feeling absolutely and from a proper workout

Table 2: Example of sentiment neutralization from input sentence

The neutralization model is essentially a sentiment classification model which first detects the sentiment of the given utterance (positive/negative). This model consists of several LSTM layers and a self-attention layer. During testing, the self-attention vector is extracted as done by Xu et al. (2018) which is then inversed and discretized as follows:

$$\hat{a_i} = \begin{cases} 0, & \text{if } a_i > 0.95 * max(a) \\ 1, & \text{otherwise} \end{cases} \quad (1)$$

where a_i is the attention weight for the i^{th} word, and $max(a)$ gives the highest attention value from the current utterance. A word is filtered out if the discretized attention weight for that word is 0. The sentiment detection model architecture is shown in figure 2.

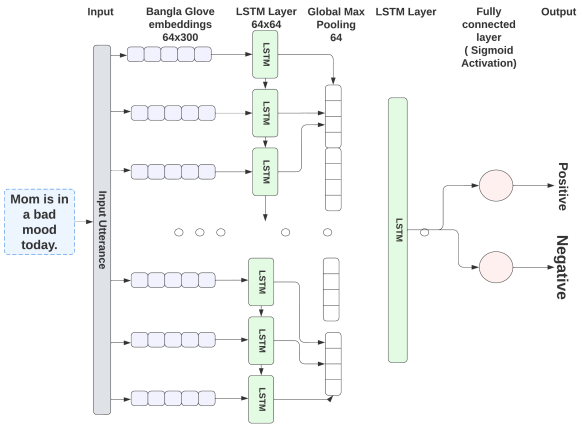

Figure 2: Sentiment detection model architecture for the Sentiment neutralization module

3.1.2 Positive Sentiment Induction

The output from the Sentiment Neutralization module is fed to the Positive Induction module as input. The module takes in a neutral utterance and incorporates positive sentiment into the utterance and returns a sentence with positive sentiment. An example is shown in table 3. For this, we use Neural Machine Translation method built on OpenNMT

framework (Klein et al., 2017) where we first train our model with a set of $< source, target >$ pairs where the source is a neutral sentence and target is its positive counter part. We use the Positive dataset provided by Mishra et al. (2019) which includes a set of positive sentences. We pass this dataset through the sentiment neutralization module to get the neutral source sentence to its positive target sentence and use these $< source, target >$ pairs to train the positive induction module. The input sentences are transformed into embeddings that go through the translation encoders and decoders. The encoders and decoders are both built with LSTM layers.

Neutral Input	Positive Output
Is feeling absolutely and from a proper workout	Is feeling absolutely amazing and high got away from a proper workout

Table 3: Example of positive sentiment induction from neutralized sentence

3.2 Retrieval of Commonsense

This module is used to retrieve additional context for the sarcastic sentence based on commonsense knowledge. Figure 3 demonstrates a schematic view of this module. We discuss the detailed process in the following sections. Additionally, we show an example input-output pair for this module in table 4.

Input	Commonsense Sentence
His presentation was bad	The manager is criticized by his boss after a presentation

Table 4: Example of commonsense sentence generation from input sentence

3.2.1 Generation of Commonsense Knowledge

For generating commonsense knowledge context, $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ (West et al., 2021) is used. First, we feed the input sentence to $\text{COMET}^{\text{DIS}}_{\text{TIL}}$. $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ is a machine trained 1.5B parameters commonsense model generated by applying knowledge distillation (Hinton et al., 2015) on a general language model, GPT-3. It offers 23 commonsense relation types. For our study, we use the **xEffect** relation. From the three variants of $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ ($\text{COMET}^{\text{DIS}}_{\text{TIL}}$, $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ + $\text{critic}_{\text{low}}$ and $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ + $\text{critic}_{\text{high}}$), we have chosen $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ + $\text{critic}_{\text{high}}$ for our work. The model

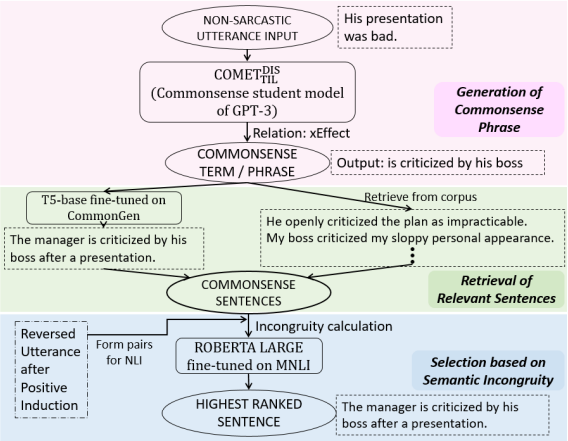

Figure 3: Model Architecture for Retrieval of Commonsense module

returns a contextual phrase pertaining to the **xEffect** relation with the extracted words of the non-sarcastic sentence. For a non-sarcastic sentence "His presentation was bad", $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ predicts the contextual phrase with **xEffect** relation – 'is criticized by his boss'.

3.2.2 Retrieval of Relevant Sentences

Once we have the inferred contextual phrase, we retrieve relevant sentences. For doing so, we imply 2 methods - 1. Retrieval from corpus and 2. Generation from the inferred phrase.

- **Retrieval from corpus:** First, from the contextual phrase, we extract the keyword. Then using the keyword, we search for related sentences in a corpus. We use Sentencedict.com[5] as the retrieval corpus. For filtering the retrieved sentences, two constraints are set - (a) the commonsense concept should appear at the beginning or at the end of the retrieved sentences; (b) to maintain consistency between the length of the non-sarcastic input and its sarcastic variant, sentence length should be less than twice the number of tokens in the non-sarcastic input. Next, we check the consistency of the pronoun in the retrieved sentence and the pronoun in the input sentence. If the pronoun does not match, we modify it to match the non-sarcastic text input. If the non-sarcastic input lacks a pronoun while the retrieved sentence does not, it is simply changed to "I". These constraints for retrieving the sentences and the assessment of grammatical consistency are done following the

[5] https://sentencedict.com/

work of Chakrabarty et al. (2020).

- **Generation from the inferred phrase:** Unlike the previous method, we keep the inferred phrase intact in this case. We first extract the *Subject* of the non-sarcastic input. If the sentence contains no *Subject*, we set it to 'I'. Then the auxiliary verb in the inferred context is checked and modified to match with that of the *Subject*. Then we feed the *Subject* and contextual phrase to a pre-trained sentence generation model[6]. The model fine-tunes Google's T5 on CommonGen (Lin et al., 2019). The model returns us a commonsense sentence based on the *Subject* and contextual inference. For example - the *Subject-inference* pair for the input "His presentation was bad" becomes ['His', 'is criticized by his boss'], and from this collection of words, the sentence "The manager is criticized by his boss after a presentation." is generated.

3.2.3 Selection based on Semantic Incongruity

The module in section 3.2.2 returns several sentences containing the context. Among them, we choose the sentence having the highest semantic incongruity with the sentence generated after the Reversal of Valence module. For calculating the semantic incongruity, following Chakrabarty et al. (2020), we have used the RoBERTa-large (Liu et al., 2019) model fine-tuned on the Multi-Genre NLI dataset (Williams et al., 2017). Considering the non-sarcastic input "His presentation was bad", the Retrieval of Relevant Sentences module yields a list of sentences such as - "The manager is criticized by his boss after a presentation", "He openly criticized the plan as impracticable", and "My boss criticized my sloppy personal appearance". From these sentences, the highest ranked sentence, "The manager is criticized by his boss after a presentation", is returned as the final output to this module as it contains the most semantic incongruity with the reversed sentence.

3.3 Emoji Prediction

In this module, we use a pre-trained emoji prediction model which is fine tuned on the CLIP (Radford et al. (2021)) deep learning model by OpenAI to predict an emoji from a given input. After concatenating the non-sarcastic input and the context retrieved from the Retrieval of Commonsense module, we predict an emoji based on this concatenated sentence. The model employs a masked self-attention Transformer as a text encoder and a ViT-B/32 Transformer architecture as an image encoder. By using a contrastive loss, these encoders are trained to optimize the similarity of (image, text) pairs. One version of the implementation used a Vision Transformer and the other a ResNet image encoder. The variation with the Vision Transformer is used in this case. The dataset[7] used for fine-tuning the model consists of two columns: raw tweets and emoji labels. The emoji labels correspond to the appropriate one among a set of 32 emojis shown in figure 4.

Figure 4: Set of 32 emojis

4 Experimental Setup

The dataset, model configurations for the different modules, and the evaluation criteria for our work are all discussed in the following sub sections.

4.1 Dataset

For our experiments, we utilize the Positive and Negative sentiment corpora by Mishra et al. (2019) which contains tweets and short snippets. Tweets have been normalized by eliminating hashtags, usernames, and conducting spell checking and lexical normalization using NLTK (Loper and Bird, 2002). After filtering out sentences longer than 30 words and running them through all three modules, we get the final dataset of 2k sarcastic sentences from the Mishra et al. (2019) dataset. We have made our dataset[8] publicly available.

4.2 Model Configurations

The sentiment classification model of the neutralization module is trained on the sentiment dataset

[6] https://huggingface.co/mrm8488/t5-base-finetuned-common_gen

[7] https://huggingface.co/datasets/vincentclaes/emoji-predictor

[8] https://github.com/WrightlyRong/Sarcasm-Generation-with-Emoji

Non-Sarcastic Utterance	System	Sarcastic Utterance	Sarcasticness	Creativity	Humor	Grammaticality
Home with the flu.	Full Model	Happy to be home with the fam. Being incarcerated-under the label of being mentally ill. 😫	3.67	4.33	4	5
	Without Emoji	Happy to be home with the fam. Being incarcerated-under the label of being mentally ill.	3.67	4.33	3.67	5
	Without Context	Happy to be home with the fam. 😫	3.33	3	3	5
	R^3 (Chakrabarty et al., 2020)	Home with the not flu.	1.67	1.33	1.33	3
The boss just came and took the mac away.	Full Model	The boss just ended and took the mac away awesome. Angry is not the word for it - I was furious. 😠	5	5	4.67	4.33
	Without Emoji	The boss just ended and took the mac away awesome. Angry is not the word for it - I was furious.	4	3.67	3	4.67
	Without Context	The boss just ended and took the mac away awesome. 😠	5	5	4.67	4.33
	R^3 (Chakrabarty et al., 2020)	The boss just came and took the mac away. Angry is not the word for it - I was furious.	1.67	2.33	1.67	5
Friday nights are so boring when the boyfriend is working late and then i have to work at on saturday mornings.	Full Model	Friday nights are so cute when the boyfriend is working rearrange and then i have to work at on mornings. At least they weren't bored. 😔	4	4	3.67	4
	Without Emoji	Friday nights are so cute when the boyfriend is working rearrange and then i have to work at on mornings. At least they weren't bored.	4	4	3.67	4
	Without Context	Friday nights are so cute when the boyfriend is working rearrange and then i have to work at on mornings. 😔	4	4	3.67	4
	R^3 (Chakrabarty et al., 2020)	Friday nights are so boring when the boyfriend is working early and then I have to work at on saturday mornings. Friday saw the latest addition to darlington's throbbing night life packed to the rafters.	1.33	2	1.33	5
Just finished workin bed feeling sick.	Full Model	Just finished workin feeling good. My stomach heaved and I felt sick. 😫	5	5	4.67	5
	Without Emoji	Just finished workin feeling good. My stomach heaved and I felt sick.	5	5	4.67	5
	Without Context	Just finished workin feeling good. 😫	3	3	3	5
	R^3 (Chakrabarty et al., 2020)	Just finished workin bed feeling healthy. My stomach heaved and I felt sick.	5	4.33	4.67	5

Table 5: Score comparison among the generated outputs from the different systems (Full model, Output without context, Output without emoji and the State-of-the-art model) on four categories

given by Mishra et al. (2019) where the negative sentences are labeled as 1 and the positive sentences are labeled as 0. Each word in the input sentence is first encoded with one-hot encoding and turned into a K-dimensional embedding. Then, these embeddings go through an LSTM layer with 200 hidden units, a self-attention layer, an LSTM layer with 150 hidden units and finally a softmax layer. The classifier is trained for 10 epochs with a batch size of 32, and achieves a validation accuracy of 96% and a test accuracy of 95.7%.
The positive sentiment induction module is built on top of the OpenNMT 3.0 framework, and following Mishra et al. (2019), the embedding dimensions of the encoder and decoder is set to 500, with 2 LSTM layers each consisting of 500 hidden units. Training iteration is set to 100000 and early stopping is incorporated to prevent overfitting. After training, the model produced a corpus-BLEU score of 51.3%.

4.3 Evaluation Criteria

For evaluating the performance of our proposed architecture we incorporate Human judgement. To assess the quality of the generated dataset we compare among 4 systems.

1. **Full Model** contains all the proposed modules of the framework and generates the final dataset.

2. **Without Emoji** system includes the context sentences along with the outputs from the reversal of valence module but does not contain any emoji that goes with each sarcastic sentence.

3. **Without Context** system consists of generations from the reversal of valence module as well as emoji. It does not include any context.

4. **R^3** is the state-of-the-art sarcasm generation system proposed by Chakrabarty et al. (2020).

To assess each of the four systems, we randomly choose 100 samples from our sarcastic dataset which totals to 400 output from the four systems. We evaluate these 400 generated sentences for comparing on the basis of the 4 above mentioned systems.
Following the evaluation approach proposed by Chakrabarty et al. (2020), we evaluate the generated sentences on these criteria:

1. Sarcasticness ("How sarcastic is the output?"),

2. Creativity ("How creative is the output?"),

3. Humour ("How funny is the output?"),

4. Grammaticality ("How grammatically correct is the output?").

Previous studies on sarcasm generation have employed sarcasticness as a criterion for evaluating the effectiveness of the generated outputs (Mishra et al., 2019; Chakrabarty et al., 2020; Das et al., 2022). As sarcasm exemplifies linguistic creativity (Gerrig and Gibbs Jr, 1988), creativity has been proposed as a method for operationalizing the quality of sarcastic sentences by Skalicky and Crossley (2018). The association between humor and sarcasm is frequently mentioned in literature as well (Dress et al., 2008; Lampert and Ervin-Tripp, 2006; Leggitt and Gibbs, 2000; Bowes and Katz, 2011). The grammaticality criterion assesses the syntactic accuracy and conformity of the generated sentences.

Three human judges have been chosen to rate the outputs from the 4 systems on the 4 criteria mentioned. The label indicates a rating on a scale of 1 (not at all) to 5 (very). All 3 judges label each of the 400 sentences from the 4 systems. The human judges have been chosen based on their high efficiency in English, good grasp in understanding and differentiating between Creativity, Humor and Sarcasticness in English sentences.

To assess the inter-annotator agreement for the ratings, we incorporated the Intraclass Correlation Coefficient (ICC). ICC is a statistical measure used to assess the degree of agreement or correlation among the ratings given by different evaluators or raters for a certain category or metric. The agreement scores are shown in table 6. The ICC score ranges between 0 and 1 where a higher score indicates a greater agreement among the raters. For all the four systems evaluated in our work, the ratings by 3 judges for the 4 evaluation criteria yield ICC scores above 0.9 in each case. A score above 0.9 indicates highly consistent observations and excellent agreement among the 3 judges.

Besides, human evaluation, we also evaluate our generated data against an emoji-based sarcasm detection model trained with existing emoji-based sarcastic dataset. For this, we utilize the work of Subramanian et al. (2019) and use their proposed sarcasm detection model trained with their dataset. Their data samples were tweets with emojis scraped from Twitter and were labeled either 1 (sarcastic) or 0 (non-sarcastic). The model consists of a Bi-GRU with a text encoder and an emoji encoder. We add 2k non-sarcastic texts with our generated 2k sarcastic texts and test the model with these data. The model's performance is discussed in section 5.

System	Intraclass Correlation Coefficient (ICC)			
	S	C	H	G
Full Model	0.90	0.92	0.92	0.94
Without Emoji	0.95	0.96	0.95	0.92
Without Context	0.93	0.94	0.94	0.93
R^3 (Chakrabarty et al., 2020)	0.97	0.97	0.97	0.97

Table 6: Intraclass Correlation Coefficient (ICC) scores on different metrics for the four systems. Here, S=Sarcasticness, C=Creativity, H=Humor, G=Grammaticality are the 4 evaluation criteria.

5 Experimental Results & Analysis

System	$\text{Variance}_{\text{eval}}$			
	S	C	H	G
Full Model	0.62	0.59	0.60	0.96
Without Emoji	0.74	0.73	0.65	0.96
Without Context	0.57	0.43	0.44	1.02
R^3 (Chakrabarty et al., 2020)	1.48	1.17	1.16	0.99

Table 7: Variances among each evaluation criterion for each system. Here, S=Sarcasticness, C=Creativity, H=Humor, G=Grammaticality are the 4 evaluation criteria.

Table 5 shows the comparison between a few sample sarcastic outputs across the various systems (our full model, output without the context, output without any emoji and lastly the state-of-the-art model (Chakrabarty et al., 2020) on different measures (Sarcasticness, Creativity, Humor and Grammaticality). Each score is the average rating given by the three human judges. Table 7 shows the variances among each evaluation criterion for each of the four systems. The variances among the four criteria for the system R^3 are higher than all the other systems.

Table 8 shows the average ratings on 100 samples by human judges for generated sarcastic sentences from the four systems based on the four categories. Our full model achieves the highest average score among all the systems including the state-of-the-art sarcasm generation model by Chakrabarty et al. (2020) on three of the four categories except Grammaticality. Besides the full model, the without

System	Sarcasticness	Creativity	Humor	Grammaticality
Full Model	**3.44**	**3.29**	**3.16**	3.72
Without Emoji	2.77	2.83	2.69	3.7
Without Context	3.1	2.99	2.88	3.72
R^3 (Chakrabarty et al., 2020)	2.32	2.2	2.1	**4.29**

Table 8: Average ratings by human judges for outputs from the four systems

emoji system and without context system also outperform the state-of-the-art on Sarcasticness, Creativity and Humor. Our system lacks in Grammaticality due to the fact that we replace the rule based approach of the reversal of valence module by Chakrabarty et al. (2020) with a deep learning approach which results in a slightly more significant information loss. However, the rule based model performs worse in case of the other three categories as it fails to generalize on all types of sentence structures. It is apparent from the scores that context plays an important role in recognising a sarcastic sentence. Additionally, the notable improvement in the score for full model compared to the without emoji model suggests that emojis obviously help better detect the incongruity that exist in sarcastic utterances.

The emoji based sarcasm detection model by Subramanian et al. (2019) gives an F1-score of 67.28% and an ROC AUC score of 53.33% on our generated data samples. It is to be noted that the model's training data samples have significantly different sentence structure than the test samples.

Conclusion

We propose a novel multi-modular framework for sarcasm generation with emoji considering two key characteristics of sarcasm: reversal of valence and semantic incongruity between the sarcastic remark and the context. To generate sarcastic sentences, we first neutralize the input sentence's sentiment and then add positive sentiment to the sentence to reverse its meaning. We also incorporate a relevant emoji and its contextual information to enhance the sarcastic effect. We conclude by evaluating our model using human judgement.

Limitations

Although our proposed architecture successfully generates emoji-based sarcastic sentences from non-sarcastic texts, in some cases, particularly longer sentences, adding commonsense context does not add much to make it more sarcastic as in such cases, the longer sentences already contain the contextual information. In future, we plan to modify our architecture in a way such that it can identify whether or not adding commonsense context would be necessary.

In our work, we have used $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ to generate additional commonsense context. So the performance of our proposed architecture heavily depends on the accuracy of $\text{COMET}^{\text{DIS}}_{\text{TIL}}$. In future, we would like to find and incorporate better models for generating commonsense context.

The low grammaticality score by our final model is likely to be caused by the insufficient training data for the Positive Sentiment Induction module for which the model could not generalize properly. We believe that there is still room for improvement here by collecting and adding more training samples to improve the model's performance. To further fix the grammatical errors we plan to add another module after the Positive Induction module where the module will use a Transformer based grammar correction model which will take a sentence with bad grammar and output a grammatically correct sentence.

Lastly, our emoji prediction module only predicts one emoji per sentence. However, to make a sentence sarcastic, it is not uncommon to use more than one emoji. Hence, we plan to explore multi-label emoji prediction in the future.

References

Marc Aguert. 2022. Paraverbal expression of verbal irony: vocal cues matter and facial cues even more. *Journal of Nonverbal Behavior*, 46(1):45–70.

Silvio Amir, Byron C Wallace, Hao Lyu, and Paula Carvalho Mário J Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. *arXiv preprint arXiv:1607.00976*.

Adithya Avvaru, Sanath Vobilisetty, and Radhika Mamidi. 2020. Detecting sarcasm in conversation context using transformer-based models. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 98–103.

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, pages 574–577.

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.

Arup Baruah, Kaushik Das, Ferdous Barbhuiya, and Kuntal Dey. 2020. Context-aware sarcasm detection using bert. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 83–87.

Christos Baziotis, Nikos Athanasiou, Georgios Paraskevopoulos, Nikolaos Ellinas, Athanasia Kolovou, and Alexandros Potamianos. 2018. Ntua-slp at semeval-2018 task 2: Predicting emojis using rnns with context-aware attention. *arXiv preprint arXiv:1804.06657*.

Santosh Kumar Bharti, Korra Sathya Babu, and Sanjay Kumar Jena. 2015. Parsing-based sarcasm sentiment recognition in twitter data. In *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 1373–1380. IEEE.

Santosh Kumar Bharti, Ramkrushna Pradhan, Korra Sathya Babu, and Sanjay Kumar Jena. 2017. Sarcasm analysis on twitter data using machine learning approaches. *Trends in Social Network Analysis*, pages 51–76.

Santosh Kumar Bharti, Bakhtyar Vachha, RK Pradhan, Korra Sathya Babu, and Sanjay Kumar Jena. 2016. Sarcastic sentiment detection in tweets streamed in real time: a big data approach. *Digital Communications and Networks*, 2(3):108–121.

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

Christian Burgers, Margot Van Mulken, and Peter Jan Schellens. 2012. Verbal irony: Differences in usage across written genres. *Journal of Language and Social Psychology*, 31(3):290–310.

Marius Cristian Buzea, Stefan Trausan-Matu, and Traian Rebedea. 2022. Automatic fake news detection for romanian online news. *Information*, 13(3):151.

Yitao Cai, Huiyu Cai, and Xiaojun Wan. 2019. Multi-modal sarcasm detection in twitter with hierarchical fusion model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2506–2515.

John D Campbell and Albert N Katz. 2012. Are there necessary conditions for inducing a sense of sarcastic irony? *Discourse Processes*, 49(6):459–480.

Spencer Cappallo, Thomas Mensink, and Cees GM Snoek. 2015. Image2emoji: Zero-shot emoji prediction for visual media. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1311–1314.

Tuhin Chakrabarty, Debanjan Ghosh, Smaranda Muresan, and Nanyun Peng. 2020. R3: Reverse, retrieve, and rank for sarcasm generation with commonsense knowledge. In *Annual Meeting of the Association for Computational Linguistics*.

Dushyant Singh Chauhan, Gopendra Vikram Singh, Aseem Arora, Asif Ekbal, and Pushpak Bhattacharyya. 2022. An emoji-aware multitask framework for multimodal sarcasm detection. *Knowledge-Based Systems*, 257:109924.

Tanvi Dadu and Kartikey Pant. 2020. Sarcasm detection using context separators in online discourse. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 51–55.

Sourav Das, Soumitra Ghosh, Anup Kumar Kolya, and Asif Ekbal. 2022. Unparalleled sarcasm: a framework of parallel deep lstms with cross activation functions towards detection and generation of sarcastic statements. *Language Resources and Evaluation*, pages 1–38.

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

Yufeng Diao, Hongfei Lin, Liang Yang, Xiaochao Fan, Yonghe Chu, Kan Xu, and Di Wu. 2020. A multi-dimension question answering network for sarcasm detection. *IEEE Access*, 8:135152–135161.

Ning Ding, Sheng-wei Tian, and Long Yu. 2022. A multimodal fusion method for sarcasm detection based on late fusion. *Multimedia Tools and Applications*, 81(6):8597–8616.

Xiangjue Dong, Changmao Li, and Jinho D Choi. 2020. Transformer-based context-aware sarcasm detection in conversation threads from social media. *arXiv preprint arXiv:2005.11424*.

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.

Ibrahim Abu Farha and Walid Magdy. 2020. From arabic sentiment analysis to sarcasm detection: The arsarcasm dataset. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 32–39.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *arXiv preprint arXiv:1708.00524*.

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

Richard J Gerrig and Raymond W Gibbs Jr. 1988. Beyond the lexicon: Creativity in language production. *Metaphor and Symbol*, 3(3):1–19.

Richard J Gerrig and Yevgeniya Goldvarg. 2000. Additive effects in the perception of sarcasm: Situational disparity and echoic mention. *Metaphor and Symbol*, 15(4):197–208.

Aniruddha Ghosh and Tony Veale. 2016. Fracking sarcasm using neural network. In *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 161–169.

Aniruddha Ghosh and Tony Veale. 2017. Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 482–491.

Debanjan Ghosh, Alexander R Fabbri, and Smaranda Muresan. 2018. Sarcasm analysis using conversation context. *Computational Linguistics*, 44(4):755–792.

Hunter Gregory, Steven Li, Pouya Mohammadi, Natalie Tarn, Rachel Draelos, and Cynthia Rudin. 2020. A transformer approach to contextual sarcasm detection in twitter. In *Proceedings of the second workshop on figurative language processing*, pages 270–275.

Raj Kumar Gupta and Yinping Yang. 2017. Crystalnest at semeval-2017 task 4: Using sarcasm detection for enhancing sentiment classification and quantification. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 626–633.

Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. 2018. Cascade: Contextual sarcasm detection in online discussion forums. *arXiv preprint arXiv:1805.06413*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Suzana Ilić, Edison Marrese-Taylor, Jorge A Balazs, and Yutaka Matsuo. 2018. Deep contextualized word representations for detecting sarcasm and irony. *arXiv preprint arXiv:1809.09795*.

Tanya Jain, Nilesh Agrawal, Garima Goyal, and Niyati Aggrawal. 2017. Sarcasm detection of tweets: A comparative study. In *2017 Tenth International Conference on Contemporary Computing (IC3)*, pages 1–6. IEEE.

Nikhil Jaiswal. 2020. Neural sarcasm detection using conversation context. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 77–82.

Soroush Javdan, Behrouz Minaei-Bidgoli, et al. 2020. Applying transformers and aspect-based sentiment analysis approaches on sarcasm detection. In *Proceedings of the second workshop on figurative language processing*, pages 67–71.

Amit Kumar Jena, Aman Sinha, and Rohit Agarwal. 2020. C-net: Contextual network for sarcasm detection. In *Proceedings of the second workshop on figurative language processing*, pages 61–66.

Aditya Joshi, Diptesh Kanojia, Pushpak Bhattacharyya, and Mark Carman. 2017. Sarcasm suite: a browser-based engine for sarcasm detection and generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Aditya Joshi, Anoop Kunchukuttan, Pushpak Bhattacharyya, and Mark James Carman. 2015. Sarcasmbot: An open-source sarcasm-generation module for chatbots. In *WISDOM Workshop at KDD*.

Aditya Joshi, Vaibhav Tripathi, Kevin Patel, Pushpak Bhattacharyya, and Mark Carman. 2016. Are word embedding-based features useful for sarcasm detection? *arXiv preprint arXiv:1610.00883*.

A Kalaivani and D Thenmozhi. 2020. Sarcasm identification and detection in conversion context using bert. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 72–76.

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

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. *arXiv preprint arXiv:1701.02810*.

Akshi Kumar, Saurabh Raj Sangwan, Anshika Arora, Anand Nayyar, Mohamed Abdel-Basset, et al. 2019. Sarcasm detection using soft attention-based bidirectional long short-term memory model with convolution network. *IEEE access*, 7:23319–23328.

Amardeep Kumar and Vivek Anand. 2020. Transformers on sarcasm detection with context. In *Proceedings of the second workshop on figurative language processing*, pages 88–92.

Avinash Kumar, Vishnu Teja Narapareddy, Pranjal Gupta, Veerubhotla Aditya Srikanth, Lalita Bhanu Murthy Neti, and Aruna Malapati. 2021. Adversarial and auxiliary features-aware bert for sarcasm detection. In *8th ACM IKDD CODS and 26th COMAD*, pages 163–170.

Avinash Kumar, Vishnu Teja Narapareddy, Veerubhotla Aditya Srikanth, Aruna Malapati, and Lalita Bhanu Murthy Neti. 2020. Sarcasm detection using multi-head attention based bidirectional lstm. *Ieee Access*, 8:6388–6397.

Martin D Lampert and Susan M Ervin-Tripp. 2006. Risky laughter: Teasing and self-directed joking among male and female friends. *Journal of Pragmatics*, 38(1):51–72.

Hankyol Lee, Youngjae Yu, and Gunhee Kim. 2020. Augmenting data for sarcasm detection with unlabeled conversation context. *arXiv preprint arXiv:2006.06259*.

John S Leggitt and Raymond W Gibbs. 2000. Emotional reactions to verbal irony. *Discourse processes*, 29(1):1–24.

Bin Liang, Chenwei Lou, Xiang Li, Min Yang, Lin Gui, Yulan He, Wenjie Pei, and Ruifeng Xu. 2022. Multi-modal sarcasm detection via cross-modal graph convolutional network. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1767–1777.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2019. Commongen: A constrained text generation challenge for generative commonsense reasoning. *arXiv preprint arXiv:1911.03705*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. *arXiv preprint cs/0205028*.

Chenwei Lou, Bin Liang, Lin Gui, Yulan He, Yixue Dang, and Ruifeng Xu. 2021. Affective dependency graph for sarcasm detection. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1844–1849.

Navonil Majumder, Soujanya Poria, Haiyun Peng, Niyati Chhaya, Erik Cambria, and Alexander Gelbukh. 2019. Sentiment and sarcasm classification with multitask learning. *IEEE Intelligent Systems*, 34(3):38–43.

Aaron Maladry, Els Lefever, Cynthia Van Hee, and Veronique Hoste. 2022. 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.

Diana G Maynard and Mark A Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Lrec 2014 proceedings*. ELRA.

Abhijit Mishra, Tarun Tater, and Karthik Sankaranarayanan. 2019. A modular architecture for unsupervised sarcasm generation. 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 6144–6154.

Abdullah Y Muaad, Hanumanthappa Jayappa Davanagere, JV Benifa, Amerah Alabrah, Mufeed Ahmed Naji Saif, D Pushpa, Mugahed A Al-Antari, and Taha M Alfakih. 2022. Artificial intelligence-based approach for misogyny and sarcasm detection from arabic texts. *Computational Intelligence and Neuroscience*, 2022.

Shubhadeep Mukherjee and Pradip Kumar Bala. 2017. Detecting sarcasm in customer tweets: an nlp based approach. *Industrial Management & Data Systems*.

Usman Naseem, Imran Razzak, Peter Eklund, and Katarzyna Musial. 2020. Towards improved deep contextual embedding for the identification of irony and sarcasm. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Silviu Oprea, Steven Wilson, and Walid Magdy. 2021. Chandler: An explainable sarcastic response generator. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 339–349.

Shereen Oraby, Vrindavan Harrison, Amita Misra, Ellen Riloff, and Marilyn Walker. 2017. Are you serious?: Rhetorical questions and sarcasm in social media dialog. *arXiv preprint arXiv:1709.05305*.

Reynier Ortega-Bueno, Paolo Rosso, and José E Medina Pagola. 2022. Multi-view informed attention-based model for irony and satire detection in spanish variants. *Knowledge-Based Systems*, 235:107597.

Rajnish Pandey and Jyoti Prakash Singh. 2023. Bert-lstm model for sarcasm detection in code-mixed social media post. *Journal of Intelligent Information Systems*, 60(1):235–254.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, and Prateek Vij. 2016. A deeper look into sarcastic tweets using deep convolutional neural networks. *arXiv preprint arXiv:1610.08815*.

Rolandos-Alexandros Potamias, Georgios Siolas, and Andreas Stafylopatis. 2019. A robust deep ensemble classifier for figurative language detection. In *International Conference on Engineering Applications of Neural Networks*, pages 164–175. Springer.

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.

Anukarsh G Prasad, S Sanjana, Skanda M Bhat, and BS Harish. 2017. Sentiment analysis for sarcasm detection on streaming short text data. In *2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA)*, pages 1–5. IEEE.

Tomáš Ptáček, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on czech and english twitter. In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers*, pages 213–223.

Veranika Puhacheuskaya and Juhani Järvikivi. 2022. I was being sarcastic!: The effect of foreign accent and political ideology on irony (mis) understanding. *Acta Psychologica*, 222:103479.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Yafeng Ren, Donghong Ji, and Han Ren. 2018. Context-augmented convolutional neural networks for twitter sarcasm detection. *Neurocomputing*, 308:1–7.

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.

Jie Ruan, Yue Wu, Xiaojun Wan, and Yuesheng Zhu. 2022. How to describe images in a more funny way? towards a modular approach to cross-modal sarcasm generation. *arXiv preprint arXiv:2211.10992*.

Edoardo Savini and Cornelia Caragea. 2022. Intermediate-task transfer learning with bert for sarcasm detection. *Mathematics*, 10(5):844.

Rossano Schifanella, Paloma De Juan, Joel Tetreault, and Liangliang Cao. 2016. Detecting sarcasm in multimodal social platforms. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 1136–1145.

Boaz Shmueli, Lun-Wei Ku, and Soumya Ray. 2020. Reactive supervision: A new method for collecting sarcasm data. *arXiv preprint arXiv:2009.13080*.

Stephen Skalicky and Scott Crossley. 2018. Linguistic features of sarcasm and metaphor production quality. In *Proceedings of the Workshop on Figurative Language Processing*, pages 7–16.

Himani Srivastava, Vaibhav Varshney, Surabhi Kumari, and Saurabh Srivastava. 2020. A novel hierarchical bert architecture for sarcasm detection. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 93–97.

Jayashree Subramanian, Varun Sridharan, Kai Shu, and Huan Liu. 2019. Exploiting emojis for sarcasm detection. In *International conference on social computing, behavioral-cultural modeling and prediction and behavior representation in modeling and simulation*, pages 70–80. Springer.

Yi Tay, Luu Anh Tuan, Siu Cheung Hui, and Jian Su. 2018. Reasoning with sarcasm by reading in-between. *arXiv preprint arXiv:1805.02856*.

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

Zhiyuan Wen, Lin Gui, Qianlong Wang, Mingyue Guo, Xiaoqi Yu, Jiachen Du, and Ruifeng Xu. 2022. Sememe knowledge and auxiliary information enhanced approach for sarcasm detection. *Information Processing & Management*, 59(3):102883.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena D Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2021. Symbolic knowledge distillation: from general language models to commonsense models. *arXiv preprint arXiv:2110.07178*.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

Deirdre Wilson. 2006. The pragmatics of verbal irony: Echo or pretence? *Lingua*, 116(10):1722–1743.

Chuhan Wu, Fangzhao Wu, Sixing Wu, Zhigang Yuan, Junxin Liu, and Yongfeng Huang. 2018. Thu_ngn at semeval-2018 task 2: Residual cnn-lstm network with attention for english emoji prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 410–414.

Jingjing Xu, Xu Sun, Qi Zeng, Xuancheng Ren, Xiaodong Zhang, Houfeng Wang, and Wenjie Li. 2018. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. *arXiv preprint arXiv:1805.05181*.

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.

Yazhou Zhang, Dan Ma, Prayag Tiwari, Chen Zhang, Mehedi Masud, Mohammad Shorfuzzaman, and Dawei Song. 2023. Stance-level sarcasm detection with bert and stance-centered graph attention networks. *ACM Transactions on Internet Technology*, 23(2):1–21.

A Appendix

Table 9: Summary of sarcasm detection datasets from different social media platforms

	Dataset					Annotation		
	Short Text	**Long Text**	**Image**	**Samples**	**Platform**	**Manual**	**Hashtag**	**None**
(Filatova, 2012)		✓		1254	Amazon	✓		
(Riloff et al., 2013)	✓			1600	Twitter	✓		
(Ptáček et al., 2014)	✓			920000	Twitter	✓	✓	
(Barbieri et al., 2014)	✓			60000	Twitter		✓	
(Bamman and Smith, 2015)	✓			19534	Twitter		✓	
(Amir et al., 2016)	✓			11541	Twitter		✓	
(Bharti et al., 2016)	✓			1.5M	Twitter			✓
(Joshi et al., 2016)	✓			3629	Goodreads		✓	
(Ghosh and Veale, 2016)	✓			41000	Twitter		✓	
(Poria et al., 2016)	✓			100000	Twitter	✓	✓	
(Schifanella et al., 2016)	✓		✓	600925	Instagram, Tumblr, Twitter		✓	
(Zhang et al., 2016)	✓			9104	Twitter		✓	
(Felbo et al., 2017)	✓			1.6B	Twitter			✓
(Ghosh and Veale, 2017)	✓			41200	Twitter	✓		
(Khodak et al., 2017)	✓			533.3M	Reddit	✓		
(Oraby et al., 2017)		✓		10270	Debate forum	✓	✓	
(Prasad et al., 2017)	✓			2000	Twitter	✓		
(Baziotis et al., 2018)	✓			550M	Twitter			✓
(Hazarika et al., 2018)	✓			219368	Reddit	✓		
(Ghosh et al., 2018)	✓	✓		36391	Twitter, Reddit, Discussion Forum	✓	✓	
(Ilić et al., 2018)	✓	✓		419822	Twitter, Reddit, Debate Forum	✓	✓	

(Tay et al., 2018)	✓	✓		94238	Twitter, Reddit, Debate Forum	✓	✓	
(Van Hee et al., 2018)	✓			4792	Twitter	✓	✓	
(Wu et al., 2018)	✓			4618	Twitter	✓	✓	
(Majumder et al., 2019)	✓			994	Twitter		✓	
(Cai et al., 2019)			✓	24635	Twitter		✓	
(Kumar et al., 2019)	✓	✓		24635	Twitter, Reddit, Debate Forum		✓	
(Subramanian et al., 2019)	✓	✓		12900	Twitter, Facebook		✓	
(Jena et al., 2020)	✓			13000	Twitter, Reddit	✓	✓	
(Potamias et al., 2020)	✓			533.3M	Twitter, Reddit	✓	✓	

Table 10: Performance summary of various approaches used in sarcasm detection

	Data	**Architecture**	**Performance**			
			Accuracy	**F1-Score**	**Precision**	**Recall**
(Davidov et al., 2010)	Tweets	SASI (Semi-supervised Algorithm for Sarcasm Identification)	0.896	0.545	0.727	0.436
(Gupta and Yang, 2017)	Tweets	CrystalNet		0.60	0.52	0.70
(Bharti et al., 2017)	Tweets	PBLGA with SVM		0.67	0.67	0.68
(Mukherjee and Bala, 2017)	Tweets	Naive Bayes	0.73			
(Jain et al., 2017)	Tweets	Weighted Ensemble	0.853		0.831	0.298
(Poria et al., 2016)	Tweets	CNN-SVM		0.9771		
(Ghosh and Veale, 2016)	Tweets	CNN-LSTM-DNN		0.901	0.894	0.912
(Zhang et al., 2016)	Tweets	GRNN	0.9074	0.9074		
(Oraby et al., 2017)	Tweets	SVM + W2V + LIWC		0.83	0.80	0.86
(Hazarika et al., 2018)	Reddit posts	CASCADE	0.79	0.86		
(Ren et al., 2018)	Tweets	CANN-KEY		0.6328		
		CANN-ALL		0.6205		

(Tay et al., 2018)	Tweets, Reddit posts	MIARN	Twitter: 0.8647	0.86	0.8613	0.8579
			Reddit: 0.6091	0.6922	0.6935	0.7005
(Ghosh et al., 2018)	Reddit posts	multiple-LSTM	0.7458	0.7607		0.7762
(Diao et al., 2020)	Internet arguments	MQA (Multi-dimension Question Answering model)		0.762	0.701	0.835
(Kumar et al., 2020)	Reddit posts	MHA-BiLSTM		0.7748	0.7263	0.8303
(Kumar et al., 2019)	Tweets	sAtt-BiLSTM convNet	0.9371			
(Majumder et al., 2019)	Text snippets	Multi task learning with fusion and shared attention		0.866	0.9101	0.9074
(Potamias et al., 2019)	reviews of laptops and restaurants	DESC (Deep Ensemble Soft Classifier)	0.74	0.73	0.73	0.73
(Srivastava et al., 2020)	Tweets, Reddit posts	BERT + BiLSTM + CNN	Twitter: 0.74			
			Reddit: 0.639			
(Gregory et al., 2020)	Tweets, Reddit posts	Transformer ensemble (BERT, RoBERTa, XLNet, RoBERTa-large, and ALBERT)		0.756	0.758	0.767
(Potamias et al., 2020)	Tweets, Reddit politics	RCNN-RoBERTa	Twitter: 0.91	0.90	0.90	0.90
			Reddit: 0.79	0.78	0.78	0.78
(Javdan et al., 2020)	Tweets	LCF-BERT		0.73		
	Reddit posts	BERT-base-cased		0.734		
(Lee et al., 2020)	Tweets, Reddit posts	BERT + BiLSTM + NeXtVLAD	Twitter	0.8977	0.8747	0.9219
			Reddit	0.7513	0.6938	0.8187
(Baruah et al., 2020)	Tweets, Reddit posts	BERT-large-uncased	Twitter	0.743	0.744	0.748
			Reddit	0.658	0.658	0.658

(Avvaru et al., 2020)	Tweets, Reddit posts	BERT	Twitter	0.752		
			Reddit	0.621		
(Jaiswal, 2020)	Tweets, Reddit posts	Ensemble of several combinations of RoBERTa-large		0.790	0.790	0.792
(Shmueli et al., 2020)	Tweets	BERT	0.703	0.699	0.70 0.7741	
(Dadu and Pant, 2020)	Tweets, Reddit posts	RoBERTa-large	Twitter	0.772	0.772	0.772
			Reddit	0.716	0.716	0.718
(Kalaivani and Thenmozhi, 2020)	Tweets, Reddit posts	BERT	Twitter	0.722	0.722	0.722
			Reddit	0.679	0.679	0.679
(Naseem et al., 2020)	Tweets	T-DICE + BiLSTM + ALBERT	0.93	0.93		
(Dong et al., 2020)	Tweets, Reddit posts	context-aware RoBERTa-large	Twitter	0.783	0.784	0.789
			Reddit	0.744	0.745	0.749
(Kumar and Anand, 2020)	Tweets, Reddit posts	context-aware RoBERTa-large	Twitter	0.772	0.773	0.774
			Reddit	0.691	0.693	0.699
(Kumar et al., 2021)	Tweets	AAFAB (Adversarial and Auxiliary Features-Aware BERT)		0.7997	0.8101	0.7896
(Lou et al., 2021)	Tweets, Reddit posts	ADGCN-BERT (Affective Dependency Graph Convolutional Network)	Twitter: 0.9031	0.8954		
			Reddit: 0.8077	0.8077		