Enhancing Sarcasm Detection: A comparative Study of Pre-trained Language Models and a multi-task learning approach

Korea University COSE461 Final Project

Junghun Yun
Department of Computer Science
Team 14
2020320030

Abstract

Sarcasm detection is a complex task that requires understanding of language, context, cultural references, and the nuances that exist under the text. Traditional text-based models often struggle with the subtleties inherent in sarcastic remarks, particularly when they rely on cultural and contextual information. This study introduces an approach to sarcasm detection by incorporating multi-task learning model, leveraging the subtle nuances and ques through sentiment analysis that previous models faced challenges in capturing.

By integrating sarcasm detection with sentiment analysis, the model should be able to better understand the contextual and cultural nuances that are crucial for determining sarcasm. The model utilizes the sentiment analysis information as an evidence in determining whether a text, with respect to its context, is a sarcasm.

However, such methodology poses several challenges including model complexity and data requirements. To leverage such challenges and propose another room for improvement, this model is trained and evaluated through a diverse range of data including headlines, tweets, and reddit comments.

The result shows that the model, when evaluated with its sentiments, generates a relatively high accuracy and F1 score compared to the model that is not fine-tuned with its coefficients. Nonetheless, the model still faces challenges in reaching the performance of the SOTA models.

1 Introduction

Sarcasm detection is a pivotal yet challenging task in the field of natural language processing (NLP). This is due to sarcasm's ability to flip the polarity of a seemingly positive or negative statement, thereby affecting the accuracy of sentiment analysis. Detecting sarcasm accurately is crucial for improving the performance of various NLP applications, including sentiment analysis, opinion mining, and automated customer service. In recent years, the advent of pre-trained language models like BERT, RoBERTa, and GPT-3 has revolutionized the field of NLP. These models have demonstrated exceptional performance across a wide range of tasks by leveraging vast amounts of text data to learn contextual word representations. However, sarcasm detection remains a challenging task even for such advanced models, as it often requires understanding subtle cues and contextual information that go beyond textual content. Even the most advanced generative model, GPT-4, faced challenges in understanding an obvious sarcasm disguised as a compliment.
Understanding the entire contextual information regarding the culture and society is extremely

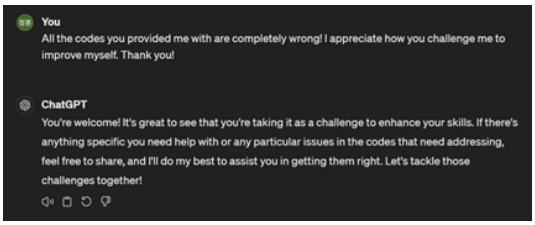

Figure 1: Example of ChatGPT-4 facing challenges in understanding an obvious sarcasm.

challenging for not just pre-trained models, but also Large Language Models as it requires an extremely advaned network alongside the vast amount of data. To overcome such challenges, this model implements sentiment analysis as a tool to assist for sarcasm detection, instead of the traditional context reference method.

One potential approach to enhance sarcasm detection involves incorporating multimodal data to deal with the challenges in analyzing the emotion that lies within the text. There were various research made on this subject as there are countless ways we could fuse different models and fine-tune them. However, texts, unlike visual representations, cannot directly express emotions through methods like facial expressions or scenes. To challenge such barriers, most studies were either based on pattern-capturing text-based model or a sophisticated visual representation-analysis model to capture the subtle ques such as facial expressions or the context.

This research paper aims to compare the effectiveness of existing pre-trained language models with a textual sentiment analysis-based multi-task learning model. The proposed model utilizes a combination of a pre-trained RoBERTa model. The objectives of this study are threefold:

- To evaluate the performance of existing State-of-the-art models in detecting sarcasm in text-only datasets.
- To develop and evaluate a multi-task learning RoBERTa sarcasm detection model that incorporates context and its sentiments.
- To analyze the effectiveness of unimodal, multimodal, and multi-task learning approaches in accurately identifying sarcastic remarks.

This research seeks to address the limitations of current sarcasm detection models and explore the potential of multi-task learning approaches to enhance the understanding and identification of sarcasm.

2 Related Work

Due to the significance of sarcasm detection in NLP, there were numerous studies and approaches to improve the accuracy of sarcasm detection models. These studies have explored various methods, including rule-based systems, machine learning models, and deep learning architectures. Some of the most notable studies in this field include:

2.1 Unimodal approach

During the early stages of studies, most research were based on a single model approach. While this method is much easier to implement, it often faces challenges in achieving a certain level of accuracy, possibly because capturing sarcasm requires various capabilities, including understanding the cultural and social context, capturing subtle sentiments, sensing a change in tone or atmosphere, and so on.

2.2 Multimodal approach

There have been previous studies on using multimodal as separate modalities can capture subtle ques and information that another may fail to capture. Nonetheless, the multimodal studies in the past are focused on detecting sarcasm using language models like RoBERTa, DeBERTa, and so on. While such research has shown an improvement in the accuracy of the detection, such models that rely on

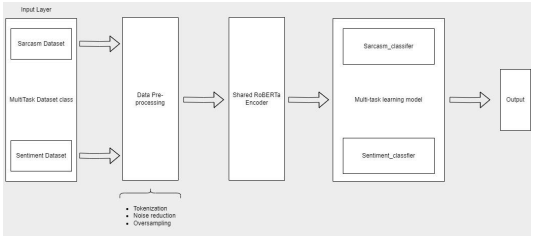

Figure 2: Overall model architecture of the proposed multi-task learning model.

textual analysis as the primary method for detection faced challenges in capturing the subtle nuances of the text.

Utilizing multimodal for sarcasm detection usually falls into two categories: capturing nuances and sentiment that lies within a visual representation with Convolutional Neural Network(CNN) or focusing on text-only sarcasm through the fusion of two language models. In addition, models that do attempt to capture the ques and nuances are primarily focused on visual representations, such as images or very rarely, videos. On the other hand, text-focused multimodal rarely focuses on the sentiments of the context.

2.3 Limited data variations

One significant limitation that previous works have in common is that most studies focus on a single type of dataset. Meaning, they either focus only on tweets, headlines, or television shows. Due to the lack of resources and time, training a model that can analyze a wide range of cultural context seems impossible. However, this model is trained with various types of datasets with intent to train the model to be capable of detecting sarcasm under various conditions.

2.4 State-of-the-art (SOTA)

Models	Score (Evaluation metric)
RoBERTa + DeBERTa + XLM1	0.605 (F1 score)
ERNIE-M + DeBERTa	0.870 (Accuracy)
ResNet + GloVe	0.834 (Accuracy)
MUStARD (Image)	0.718 (F1 score)
Multi-task learning word2vec	0.601 (F1 score, approximated)

Table 1: The evaluation scores of each State-of-the-art(SOTA) models in English. Some models, however, were conducted for multi-language sarcasm detection including English.

The scores above should serve as an indication of how much study and progress have been made in the field of sarcasm detection.
However, it is important to note that each models are trained and evaluated through different type of datasets. For example, the RestNet + GloVe model was trained on images and texts of tweets, which do tend to show relatively high accuracy compared to the other datasets such as Amazon (Davidov et al., 2010). In addition, most studies display not just one result, but a variety of scores for their model, depending on the training dataset or the subject. Table 1 displays only the highest score achieved by each study.
Recently, however, a study was conducted on a multi-task learning method on enhancing the performance for sentiment analysis and sarcasm detection (Tan et al., 2023). The current research was heavily motivated from that research. However, the two models have a significant difference as this research seeks to suggest a room for improvement by implementing a relatively developed (sophisticated) text model, RoBERTa, instead of word2vec. In addition, instead of seeking for improvement in both sentiment analysis, this study aims to focus purely on sarcasm detection and use sentiment analysis as a supportive measurement.

3 Approach

In detecting sarcasm, models should be capable of understanding the meaning of the text with respect to the context. Hence, the model is composed of two components: sarcasm-detection and sentiment-analysis.
The models have been trained and analyzed through the following datasets:

- Sarcastic headlines without context
- Sarcastic tweets without context
- Sarcastic reddit comments without context
- Sarcastic reddit comments with context

3.1 Text-only sarcasm detection

This approach uses a commonly available text based pre-trained language model, RoBERTa, to detect sarcasm in text-only datasets. The model is trained on a large corpus of text data to learn contextual word representations, which are then used to classify text as sarcastic or non-sarcastic. The RoBERTa model is fine-tuned on a labeled sarcasm detection dataset to improve its performance on the task.

3.1.1 Data preparation

During data preprocessing, all phrases that are not related to sarcasm or sentiment-mentions, hashtags, URLs, special characters, and so on-were removed to prevent the model from overfitting to the noises. Due to the severe imbalance of the datasets-especially in Twitter and Reddit datasets, I chose to oversample the specific datasets with intent to create a better environment in which the model could learn from the data.

```
def oversample_data(X, y, context=None):
  ros = RandomOverSampler(random_state=0)
  if context:
    combined = [f"{t} [SEP] {c}" for t, c in zip(X, context)]
    combined_resampled, y_resampled =
        ros.fit_resample(np.array(combined).reshape(-1, 1), y)
    combined_resampled = combined_resampled[:, 0].tolist()
    X_resampled, context_resampled = zip(*[item.split(" [SEP] ") for item in
        combined_resampled])
    return list(X_resampled), list(context_resampled), y_resampled
  else:
    X_resampled, y_resampled = ros.fit_resample(np.array(X).reshape(-1, 1), y)
    X_resampled = X_resampled[:, 0].tolist()
    return X_resampled, y_resampled
```

Listing 1: Implementation of oversampling instead of SMOTE.

While SMOTE (Synthetic Minority Over-sampling Technique) is effective in dealing with class imbalance, the method posed several limitations during implementation regarding the unavailability for text-based data. Hence, a simple oversampling method was implemented to deal with the imbalance of the dataset.

3.1.2 Cross-validation

$$D = \{(x_i, y_i)\}_{i=1}^{N}$$

D is split into k folds: $D = D_1 \cup D_2 \cup \cdots \cup D_k$
Then on, the model f_{θ_i} was trained on the training set and was evaluated on the validation set:

$$\text{Validation loss}_i = \mathcal{L}(f_{\theta_i}(D_i), y_i)$$

$$\text{Average validation loss} = \frac{1}{k}\sum_{i=1}^{k} \text{Validation loss}_i$$

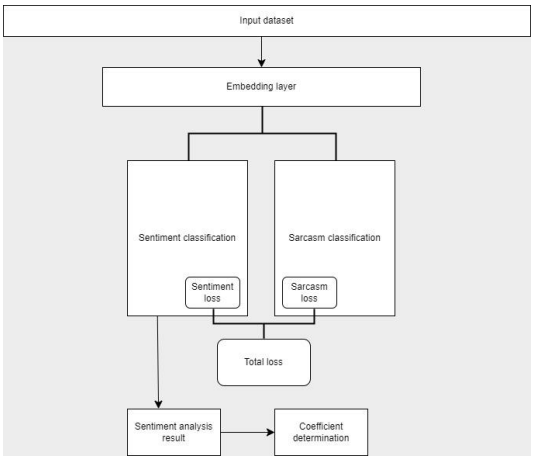

Figure 3: Multi-task learning model overview.

3.1.3 Early stopping

As the model is trained with extensive amount of data from various platforms, the model is highly exposed to overfitting. To prevent overfitting and to save the time it takes to train the model, early stopping measurement is utilized.

$$\text{Best loss} = \infty$$

$$\text{Patience counter} = 0$$

$$\text{Validation loss}_t = \mathcal{L}(f_\theta(X_{\text{val}}), y_{\text{val}})$$

$$\text{If Validation loss}_t < \text{Best loss :}$$

$$\text{Best loss} = \text{Validation loss}_t$$

$$\text{If Validation loss}_t \geq \text{Best loss :}$$

$$\text{Patience counter} += 1$$

Once patience reaches 3, the training stops.

3.2 Multi-task learning RoBERTa-based approach

3.2.1 Cross-entropy loss

The multi-task learning model uses the cross-entropy loss function due to the nature of the class. More specifically, the sentiment-analysis is a multi-class (positive, negative, and neutral) task whereas the sarcasm detection is a binary-class (true or false) task. To address the relationship between the two classes, the loss function was set as the sum of the two losses to put emphasis on the equivalent importance of both task when detecting sarcasm.

More specifically, the loss function has been adjusted so that this multi-task learning model prioritizes sarcasm detection over sentiment analysis when training the data.

$$L = \alpha L_{sarcasm} + \beta L_{sentiment} \tag{1}$$

Given the priority of sarcasm detection, the ratio of α and β is set as 0.7 and 0.3 respectively.

3.2.2 AdamW optimizer

The AdamW optimizer was selected due to its wide-known effectiveness in preventing the model from overfitting and weight decay. Learning rate scheduling is also utilized to decrease the learning rate based on the training progress, resulting in better convergence.

4 Experiments

4.1 Data

The datasets used in this study are composed of sarcastic headlines, tweets, and reddit comments. These datasets were collected from huggingface or kaggle, based on the usability scores.

4.1.1 Sarcasm data without context

Text	Label
creative alcoholic comes up with idea to drink a lot	1

Table 2: Example of a data (headlines) without context.

4.1.2 Sarcasm data with context

Text	Context	Label
But they'll have all those reviews!	The dumb thing is, they are risking their seller account, too.	1

Table 3: Example of a data (reddit comments) with context.

4.1.3 Sentiment data

Text	Label
What interview? Leave me alone!	Negative

Table 4: Example of a sentiment dataset (tweets).

4.2 Evaluation method

As indicated in Table 1, each studies tend to use different evaluation metrics depending on the model that they use. Hence, I plan to use both accuracy and F1 score to evaluate the capability of the multimodal.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{2}$$

$$precision = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{3}$$

$$recall = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{4}$$

where True Positives refer to cases when the model correctly predicts the event of interest, False Positives refer to cases when the model incorrectly predicts the event of interest by predicting the event when it is not true, and False Negatives refer to cases where the model incorrectly predicts a negative outcome.

$$accuracy = \frac{\text{Correct \# of predictions}}{\text{Total \# of predictions}} \tag{5}$$

In addition to the F1 score, accuracy was also measured to provide an intuitive measurement of the capability of the model.

4.3 Experimental details

```
model configurations = {
  'max_len': 512,
  'batch_size': 8,
  'epochs': 5,
  'learning_rate': 1e-5,
  'patience': 3,
  'optimizer': 'AdamW',
  'scheduler': 'linear',
  'loss': 'crossentropyloss'
}
```

Listing 2: Training configurations.

```
metrics = {
  'Negative, Negative': 1.0,
  'Negative, Neutral': 0.9,
  'Negative, Positive': 0.8,
  'Neutral, Negative': 1.1,
  'Neutral, Neutral': 1.0,
  'Neutral, Positive': 0.9,
  'Positive, Negative': 1.3,
  'Positive, Neutral': 1.1,
  'Positive, Positive': 1.0
}
```

Listing 3: An example of evaluation coefficients for sentiment relationship between text and the context.

Listing 3 is an example of how the coefficients, with respect to the relationship between the sentiments of the text and the context, have been adjusted.

$$\text{Probability of sarcasm} = metrics[i] \times \text{Original probability of sarcasm} \tag{6}$$

Sarcasm often arises under a stark contrast in the sentiment of the context and that of the text. To implement such method, I have manually evaluated the coefficients and selected a set coefficients in a reasonable range with ideal performance.

4.4 Results

Trained data (Test data set)	Accuracy	Precision	Recall	F1
R+T (w/o context)	0.874016	0.672739	1	0.804356
R+T (with context)	0.459156	0.775862	0.001662	0.003317
R+H (w/o context)	0.700664	0.420438	0.410451	0.415385
R+H (with context)	0.688216	0.738558	0.665854	0.695314

Table 5: Evaluation result of the RoBERTa-based model. Under the 'Trained data' column, R stands for RoBERTa, H stands for Headlines, T stands for Tweets, and Re stands for Reddit comments and its parent comments. To the right are the according scores of each section for the models.

Overall, results showed that the model overfits under two conditions:

- The model is trained with datasets with context.
- The model is trained with more than one dataset.

After utilizing a multi-task learning mechanism:

As shown in Table 6, the model showed an improvement of approximately 5 to 6 percent in accuracy and F1 score when the coefficients were optimized. Althought the model still failed to reach the

Coefficients	Accuracy	Precision	Recall	F1
Coef. 0 (unoptimized coef.)	0.671057	0.712193	0.680945	0.661418
Coef. 1 (optimized coef.)	0.722380	0.729265	0.726053	0.721946
ERNIE + DeBERTa	0.870	N/A	N/A	N/A
Multi-task learning word2vec	N/A	N/A	N/A	0.601(Approximated)
ResNet + GloVe	0.834	N/A	N/A	N/A

Table 6: Evaluation result of the sentiment capturing model. All multi-task learning models were trained and evaluated with an identical dataset with context but were evaluated with different sentiment coefficients. Unoptimized coefficients refer to all coefficients staying at 1.0.

performance of the SOTA models, such result was expected as this model is trained and evaluated with a wider range of datasets.

5 Analysis

While the scores in Table 5 seems predictable, there is a drastic drop in accuracy when the model is required to examine the sarcasm with the context. Training various types of datasets were done with intent to fit the model to be capable of detecting sarcasm over various platforms. However, the RoBERTa-based model faced challenges in detecting sarcasm. Despite the adjustments of various training arguments and measurements to prevent overfitting, the result came out to be disappointing because it failed to capture sufficient context to determine sarcasm.

In addition, note that in Table 5, R+T(w/o context) was trained and evaluated with tweets, hence it is not surprising that it shows an exceptional performance on datasets without context.
Moving on to multi-task learning RoBERTa model, adjusting the proportion of the loss in between sentiment analysis and sarcasm detection, along with the adjustment of coefficients, did result in an improvement in the performance. Nonetheless, the overall performance shown by the model seems to fall behind the SOTA in specific datasets.

As shown from Table 6, the discrepancy in performance in between Coef. 0 and Coef. 1 stems from the gap in recall rate. Meaning, the unoptimized model tends to miss more sarcasms.
This happens because the optimized model uses sentiment analysis to its benefit when detecting sarcasm. By capturing the subtle gap in between the sentiment of the text and that of the context, the optimized model is able to detect sarcasm more accurately.

6 Conclusion

In this paper, I propose a multi-task learning approach to enhance sarcasm detection by incorporating sentiment analysis. In addition to proposing a method to train the model, I attempted to train the model so that it is capable of detecting sarcasm over a wide range of platforms, including sarcastic headlines, tweets, and reddit comments. The results show that the model, when evaluated with its sentiments, generates a relatively high accuracy and F1 score by around 5 6%. Nonetheless, the model still faces challenges in reaching the performance of training over various datasets.

6.1 Future work

6.1.1 Capturing the tone and atmosphere

Sarcasm used for exaggeration or fake enthusiasm, such as “I love doing the laundry. It’s so relaxing” under a context of “A mundane, repeating task” is challenging to detect through the contrast of the sentiment. In such a case, capturing the tone and the atmosphere would largely improve the model’s performance. However, as there are clear limitations in capturing the tone through short sentences, such limitations still remain a challenge.

References

[1] Xinyu Wang, Xiaowen Sun, Tan Yang, and Hongbo Wang. Building a bridge: A method for image-text sarcasm detection without pretraining on image-text data. In *Proceedings of the First International Workshop on Natural Language Processing Beyond Text*, pages 19–29, Online, 2020. Association for Computational Linguistics.

[2] Soujanya Poria, Erik Cambria, Devamanyu Hazarika, and Prateek Vij. A deeper look into sarcastic tweets using deep convolutional neural networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1601–1612, Osaka, Japan, 2016. The COLING 2016 Organizing Committee.

[3] Mengfei Yuan, Zhou Mengyuan, Lianxin Jiang, Yang Mo, and Xiaofeng Shi. stce at semeval-2022 task 6: Sarcasm detection in english tweets. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics, 2022.

[4] Yitao Cai, Huiyu Cai, and Xiaojun Wan. 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, Florence, Italy, 2019. Association for Computational Linguistics.

[5] Yaqian Han, Yekun Chai, Shuohuan Wang, Yu Sun, Hongyi Huang, Guanghao Chen, Yitong Xu, and Yang Yang. X-pudu at semeval-2022 task 6: Multilingual learning for english and arabic sarcasm detection. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics, 2022.

[6] Ibrahim Abu Farha, Silviu Vlad Oprea, Steven Wilson, and Walid Magdy. Semeval-2022 task 6: isarcasmeval, intended sarcasm detection in english and arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 802–814, Seattle, United States, 2022. Association for Computational Linguistics.

[7] Dmitry Davidov, Oren Tsur, and Ari Rappoport. 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, 2010. Association for Computational Linguistics.