IA-BERT: Context-Aware Sarcasm Detection by Incorporating Incongruity Attention Layer for Feature Extraction

Ida Ayu Putu Ari Crisdayanti
Sungkyunkwan University
Suwon, South Korea
dayu.crish@gmail.com

JinYeong Bak
Sungkyunkwan University
Suwon, South Korea
jy.bak@skku.edu

YunSeok Choi
Sungkyunkwan University
Suwon, South Korea
ys.choi@skku.edu

Jee-Hyong Lee
Sungkyunkwan University
Suwon, South Korea
john@skku.edu

ABSTRACT

Sarcasm as a form of figurative language has been widely used to implicitly convey an offensive opinion. While preliminary researches have constantly tried to identify the sarcasm lying in a text directly from tokens within the text, it is insufficient because sarcasm does not have specific vocabularies as in polarized sentences. Especially in threads or discussions, sarcasm can be identified after getting the context information from previous replies or discussions. To this end, we propose IA-BERT, a model architecture that considers contextual information to identify incongruity features that lie in sarcastic texts. IA-BERT is embedded with a feature attention layer that combines features extracted from the response alone and interactive features obtained from the context and the response. The model leverages BERT pretrained embedding and yields a performance improvement from the standard fine-tuned BERT classifier. IA-BERT also outperforms the sophisticated architecture of LCF-BERT in the accuracy and F1-score.

CCS CONCEPTS

• **Computing methodologies → Discourse, dialogue and pragmatics**;

KEYWORDS

Sarcasm Detection, Context-Aware, Incongruity Attention

ACM Reference Format:
Ida Ayu Putu Ari Crisdayanti, JinYeong Bak, YunSeok Choi, and Jee-Hyong Lee. 2022. IA-BERT: Context-Aware Sarcasm Detection by Incorporating Incongruity Attention Layer for Feature Extraction. In *The 37th ACM/SIGAPP Symposium on Applied Computing (SAC '22), April 25–29, 2022, Virtual Event,* . ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3477314.3507131

1 INTRODUCTION

Sarcasm as a form of figurative language has negatively impacted natural language processing models that are related to sentiment analysis and opinion mining. It has been widely used by social media platform users to implicitly express an offensive message. Natural language processing and natural language understanding models have been extensively improved and enhanced to identify implicit meaning such as those contained in sarcastic expressions. This type of figurative language negatively impacts the performance of natural language processing models that are based on opinion mining. Sarcastic expressions can totally flip the polarity of opinion or sentiment reflected by the text. Therefore, sarcasm detection has been further explored to improve natural language processing applications which require sentiment analysis and opinion mining [10, 16]

The attempts made by former approaches mainly leverage tokens within the text itself to identify sarcasm [11, 20, 22]. However, capturing incongruity information from the sarcastic sentence alone is not sufficient because sarcasm does not have specific vocabularies as can be easily identified in polarized sentences. In social media and discussion platforms such as Twitter and Reddit, sarcasm can be pointed out after reading the contextual information reflected in the previous reply or discussion. Therefore, it is important to consider this contextual feature to capture the incongruity of a sarcastic response.

A preliminary attempt by leveraging machine learning-based models and some auxiliary features has been conducted to solve the sarcasm detection problem [1, 9, 19–22]. However, this type of lexical feature engineering is often hard to compute and required sufficient external model (e.g POSTagger) to perform well.

To reduce the needs of lexical features, recent works use deep neural networks to automatically extract important features from the text [7, 11, 15, 17]. These approaches take the response text alone as input and use the features extracted by neural networks to predict the sarcasm probability. More recent works are based on pretrained embedding models such as BERT and RoBERTa [4, 18]. Pretrained language models have been a powerful text classifier [3, 12]. These models produce token representations with rich linguistic features since they are pretrained on a large text corpus. The pretrained models are then fine-tuned on the sarcasm prediction dataset. To train a context-aware classifier, the model is fed with input which is the concatenation of conversational context and its response [2, 13].

SAC '22, April 25–29, 2022, Virtual Event,
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-8713-2/22/04...$15.00
https://doi.org/10.1145/3477314.3507131

Context:	"ACLU on US (il) liberalism which now celebrates when mega corps tear up writers contracts in response to controversy <URL>" ". @USER The #ACLU needs to be shut down . They pander to illegal immigrants and refugees , which is not in best interest of #USA ."
Response:	"@USER @USER says the woman who isn't white and her last name isn't Smith . #immigrant"
Label:	SARCASM

(a) The response sounds sarcastic with or without the context

Context:	"New data shows 85% of humans live under a corrupt government" "15% of humans don't live under a government?"
Response:	"I know, the glass is always half empty to some people."
Label:	SARCASM

(b) The response alone does not sound sarcastic but contain sarcasm after referring to the context

Context:	"What happened when I tried to document why Trump was taken off stage by Secret Service agents <URL>" "@USER maybe if more of your colleagues had the slightest shred of integrity , the good guys like you wouldn't become scapegoats"
Response:	"@USER @USER Your idea of integrity is confirming your bias , ignoring facts , punishing press . Kinda like dictatorships ."
Label:	NOT_SARCASM

(c) The response does not sound sarcastic with or without the context

Context:	"General Buratai cried several times during funeral oration this evening . With mourning , there are no fearsome Generals , only grieving humans" "There was a female soldier who stood not far from where I was , who sobbed throughout . Eventually had to be taken away ."
Response:	"@USER this war was creates by men in Agbada and still lead by them . Its more than just a sect at war it complicated"
Label:	NOT_SARCASM

(d) The response alone sounds sarcastic but actually non sarcastic considering the context

Table 1: Examples of sarcasm detection dataset with different characteristics. (a) The response sequence alone sounds sarcastic. In this case, the incongruity information can be captured from the tokens within the response sequence. (b) The response sounds sarcastic after referring to the context. For this case, capturing relevant information from the context is important in identifying the sarcasm intended in the response sequence. (c) The response sequence is not sarcastic with or without the context. The response use a straightforward expression to convey an opinion. (d) The response alone sounds sarcastic but actually non sarcastic considering the context. Although the response may sound offensive, it is not sarcastic as there is no sarcasm intended towards the context.

However, the pretrained models are general classification architectures and can be improved to fit the task characteristics.

To gain benefit from the pretrained embedding model, our proposed method IA-BERT is a novel architecture for context-aware sarcasm detection which leverages pretrained BERT to provide embedding representation of each token in the context and response text. It consists of three attention modules that extract incongruity features for sarcasm probability prediction. The model is evaluated on two online discourse datasets, Twitter and Reddit. IA-BERT shows performance improvement over the general BERT classifier and outperforms a sophisticated classifier architecture, LCF-BERT [27].

We describe the preliminary study on the sarcasm detection task in Section 2 and present the methodology in Section 3. In Section 4, we show the superiority of our approach with experimental results. And, we describe the related work in sarcasm detection task and compare their approaches with our proposed approach in Section 5. Finally, we conclude the paper in Section 6.

2 PRELIMINARY STUDY ON THE TASK

Different from other text classification tasks such as topic detection and sentiment analysis, sarcasm detection does not have a special set of vocabulary as attribute markers. Sarcasm lies in the incongruity between the actual text and the implicit meaning it tries to convey. The expression can be used for humorous purposes. It can also be in a form of offensive message expressed in an indirect way to insult or to show irritation [6]. Sarcasm is largely context-dependent.

In capturing information from sarcastic and non-sarcastic sequences, we identify several cases in the problem domain as follows:

The response sequence alone sounds sarcastic. In Table 1a, without looking at the context sequence, we can see that the response is a sarcastic expression towards immigrants. In this case, the incongruity information can be captured from the tokens within the response sequence.

The response sounds sarcastic after referring to the context. In Table 1b, it is hard to conclude if the response is sarcastic without knowing the context. Therefore, modeling the interaction between the response sequence and its context is also important in identifying sarcasm.

Non sarcastic response. The response uses a straightforward expression. As can be seen in Table 1c, the response sequence explicitly stating disagreement towards others' opinion.

The response sounds sarcastic but actually non sarcastic considering the context. A non sarcastic sentence may sound offensive

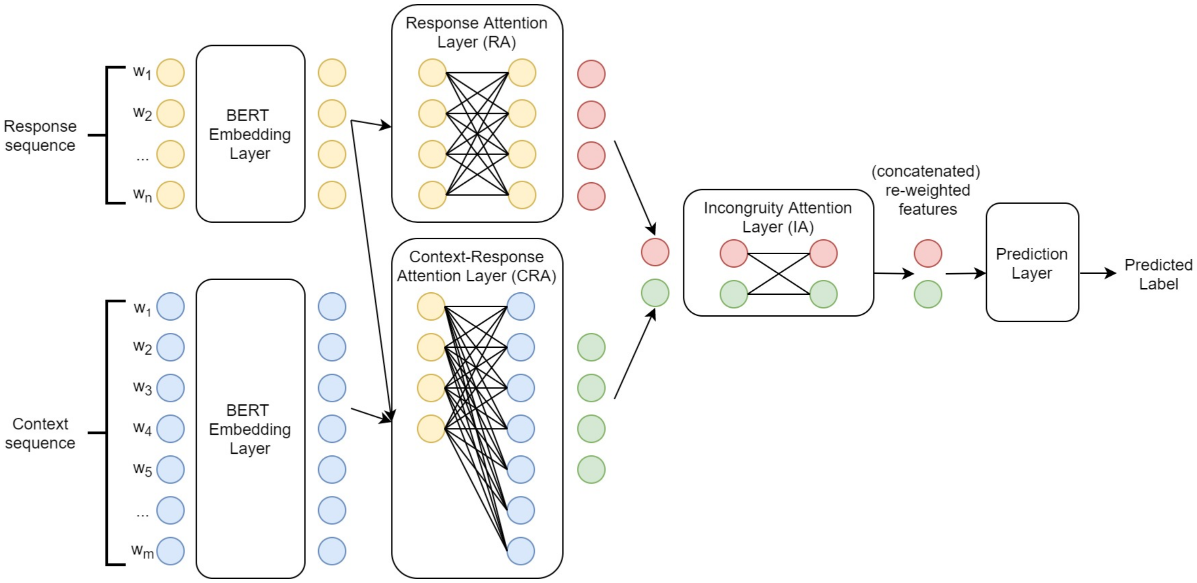
Response sequence
w1
w2
...
wn
BERT Embedding Layer
Response Attention Layer (RA)
Context sequence
w1
w2
w3
w4
w5
...
wm
BERT Embedding Layer
Context-Response Attention Layer (CRA)
Incongruity Attention Layer (IA)
(concatenated) re-weighted features
Prediction Layer
Predicted Label

Figure 1: Architecture of IA-BERT

because of the topic being discussed. This type of sentence can be found in the example in Table 1d. Although the sentence looks aggressive, it states a straightforward opinion towards the context. There is no sarcasm intended in the sentence.

Due to the above characteristics, we can conclude that considering only the response to identify sarcasm is difficult. Therefore, we need a classifier architecture which can consider both the response and the context. In this work, we propose a novel architecture for context-aware sarcasm detection called IA-BERT. As shown in Figure 1, IA-BERT first extracts response-centered features and interactive context-response features from the given context and response. These two features are then passed to a feature attention layer to learn their importance for sarcasm detection. The output vector of this layer is used to predict whether the response is sarcastic or not. The detail of each layer in the architecture is described in the following section.

3 METHODOLOGY

In context-aware sarcasm detection, the input sequences consist of context sequence and response sequence. The context sequence is a series of conversation happened before the response sequence. To perform feature extraction on the input sequences, IA-BERT is designed with an embedding layer and three attention layers. The dense layer at the end of the architecture uses the extracted features for sarcasm prediction.

3.1 BERT Embedding Layer

The BERT-shared layer is a pretrained language model that can be regarded as an embedding layer to obtain the low-dimensional vector representation of each WordPiece token [5]. To represent each token in the context and response sequences into numerical vectors, we leverage a single BERT embedding layer instead of two independent BERT layers to reduce the number of trainable parameters. It is considered that both context and response have the same syntactical and semantic characteristics. Suppose that $r = \{w_0, w_1, w_2, \ldots, w_n\}$ is an input response sequence and $c = \{w_0, w_1, w_2, \ldots, w_m\}$ is the context sequence, r^B and c^B are the embedded representations of the response and context obtained from the BERT embedding layer:

$$r^B = BERT(r), \qquad r^B \in \mathbb{R}^{d\times n}$$
$$c^B = BERT(c), \qquad c^B \in \mathbb{R}^{d\times m}$$

where d represents the BERT embedding dimension. n and m denote the maximum length of the response and context sequence respectively. These initial representations of the input are then passed through some attention modules to extract incongruity features.

3.2 Response Attention Layer (RA)

The response attention layer is based on multi-head attention mechanism [25]. We employ a self-attention on the embedded response sequence r^B to extract response centered features. The attention layer takes r^B as the key and query. Scaled dot product is applied as the attention score function as it is recommended for identical attention [27]. Suppose r^B is the input for the response attention layer, each attention head is computed as follows:

$$RA(r^B) = Softmax(\frac{QK^T}{\sqrt{d_k}}) \cdot V$$
$$Q = r^B W^q, \qquad W^q \in \mathbb{R}^{d\times d_k}$$
$$K = r^B W^k, \qquad W^k \in \mathbb{R}^{d\times d_k}$$
$$V = r^B W^v, \qquad W^v \in \mathbb{R}^{d\times d_k}$$

where the weight matrix W^q, W^k, W^v are trainable parameters and dimension d_k is equal to d/h where d is the BERT embedding dimension and h is the number of attention heads. The attention representations computed by each head are concatenated and summarized by feeding them to a linear projection layer using vector W^o. For h-heads attention layer, the response attention output will be:

$$r^{ra} = [RA_1(r^B), \dots, RA_h(r^B)] \cdot W^o$$

where $W^o \in R^{h \times d_k \times d}$ is a trainable parameter. In IA-BERT, the number of attention heads is set to eight as higher or lower number appears to reduce the model's performance.

3.3 Context-Response Attention Layer (CRA)

The context-response attention layer aims to obtain the infeaturescongruity between context sequences and response sequences in an interactive manner. The module takes the embedded representation c^B as the attention key and r^B as the query. Using the multi-head attention mechanism, we then apply the bilinear attention score function which is considered to be useful for features fusion [26]. We view this incongruity features as a fusion of information between the context and response sequences. Each attention head in this layer is computed according to the following equation:

$$\begin{aligned} &CRA(r^B, c^B) = Softmax(W^{ba} QK^T) \cdot V \\ &Q = r^B W^q, && W^q \in \mathbb{R}^{d \times d_k} \\ &K = c^B W^k, && W^k \in \mathbb{R}^{d \times d_k} \\ &V = c^B W^v, && W^v \in \mathbb{R}^{d \times d_k} \end{aligned}$$

where $W^{ba} \in \mathbb{R}^{d_k \times d_k}$ is a bilinear attention weight used to fuse the features from context and response sequence. The number of attention heads in this layer is set to eight. In this layer, all attention head representations are also concatenated and followed by a linear transformation. The output of response attention layer is denoted as $r^{cra} \in \mathbb{R}^{n \times d_o}$ where n is the length of the response sequence and d_o represents the output dimension.

3.4 Incongruity Attention Layer (IA)

From the previous two attention layers, we obtain a response-centered feature $r^{ra} \in \mathbb{R}^{n \times d_o}$ and an interactive context-response feature $r^{cra} \in \mathbb{R}^{n \times d_o}$. We then summarize each feature by performing an average pooling. The pooled representations are 300-dimensional vectors denoted as r^{ra}_{avg} and r^{cra}_{avg}. Based on our preliminary study on the task, one feature may play a more important role than the other feature depending on the cases mentioned in Section 2. The incongruity attention layer aims to perform re-weighting on these two features using the training signals obtained from the correct and incorrect predictions.

Instead of directly concatenate and flatten the features as input for the final prediction layer, we combine both features and treat them as an input sequence consisting of two elements. This representation is fed to a single-head attention layer which implements scaled dot product attention score. The attention representation is computed in the same manner as the response attention layer except for the number of attention heads.

Multi-head attention mechanism is originally proposed to learn various relations between input representations. Meanwhile, our incongruity attention layer only models the importance of two incongruity features. Therefore, we implement the single-head attention instead of multi-head attention. The re-weighted features as the output of this layer denoted as $r^{ra'}$ and $r^{cra'}$ which represent the response-centered feature and the context-response feature respectively.

Label	Dataset	
	Twitter	Reddit
SARCASM	2,500	2,200
NOT_SARCASM	2,500	2,200
Total	5,000	4,400

Table 2: Statistics of Twitter and Reddit dataset

3.5 Prediction Layer

The prediction layer takes as input the concatenation of $r^{ra'}$ and $r^{cra'}$ obtained from the incongruity attention layer. This layer consists of a linear layer and a *Softmax* function to determine the probability of each class in sarcasm detection. The probability is computed according to the following equation:

$$\hat{y} = Softmax(W^P [r^{ra'}; r^{cra'}])$$

where W^P is a trainable parameter and $\hat{y}$ is the prediction result. We use the cross-entropy loss as a training objective for the model optimization.

4 EXPERIMENTS

We evaluate IA-BERT on a publicly available training dataset from the ACL Shared Task on Figurative Language Processing [8]. We compare the performance of our best model to several baseline models on the context-aware sarcasm detection dataset.

4.1 Datasets and Hyperparameters

The shared task on sarcasm detection released the training set collected from two online discourse platforms, Twitter and Reddit. The detail of the datasets can be seen in Table 2. Using this dataset, we perform a 10-folds cross validation. We take out 10% of the data for evaluation or testing and split the remaining data with 9:1 ratio for training and validation respectively. This procedure is repeated ten times with no overlap in the test set.

From the above datasets, it was observed that 73% of the instances in the Twitter dataset have a total context-response length less than 150 and 99% of the instances in the Reddit dataset have a sequence length less than 100. To cope with this condition and the computation limit, we set the maximum length of sequence for Twitter and Reddit instances to 300 and 200 respectively. At training time, we set seed on the train-validation set split for reproducibility.

In our study, we experiment with several input sequence combinations. We train the model on the response sequence and a number of utterances in the context sequence. The setting for context sequence are last utterance, last two utterances, last three utterances, and all utterances. For context input more than one utterance, the context are concatenated in a reverse order to obtain the most recent utterance as the start of the context sequence. For the BERT baseline, the model takes the response and the context sequence as a single input by concatenating them.

Methods	Twitter				Reddit			
	Acc.	**F1**	**Prec.**	**Rec.**	**Acc.**	**F1**	**Prec.**	**Rec.**
BERT	0.795	0.798	0.786	0.814	0.684	0.693	0.675	0.719
BERT + context separator	0.795	0.794	0.798	0.795	0.674	0.634	0.730	0.567
IAN	0.538	0.526	0.512	0.536	0.500	0.667	0.500	0.100
BERT-AEN	0.727	0.749	0.694	0.821	0.552	0.576	0.548	0.630
LCF-BERT	0.800	0.804	**0.791**	0.823	0.688	0.688	0.690	**0.693**
Proposed Method	**0.805**	**0.817**	0.771	**0.870**	**0.700**	**0.696**	**0.708**	**0.693**

Table 3: Performance comparison with the baseline models on 10-folds cross validation test dataset

Input Sequence	Acc.	F1.
Response & last context	0.774	0.808
Response & last 2 contexts	**0.805**	**0.817**
Response & last 3 contexts	0.782	0.777
Response & last all contexts	0.792	0.802

(a) Twitter

Input Sequence	Acc.	F1.
Response & last context	0.680	0.673
Response & last 2 contexts	0.691	0.681
Response & last 3 contexts	**0.700**	**0.696**
Response & last all contexts	0.661	0.646

(b) Reddit

Table 4: Performance of the proposed method

Attention Modules	Acc.	F1
RA + CRA + IA	**0.805**	**0.817**
RA + CRA	0.782	0.800
CRA	0.782	0.791
RA	0.789	0.793

(a) Twitter

Attention Modules	Acc.	F1
RA + CRA + IA	**0.700**	**0.696**
RA + CRA	0.679	0.688
CRA	0.630	0.593
RA	0.685	0.638

(b) Reddit

Table 5: An ablation study on removing certain attention modules. Performance are evaluated on the Twitter and the Reddit dataset. The complete architecture of IA-BERT with 3 attention modules(layers) demonstrates a better performance than other models.

In our experiments, we use the same uncased pretrained BERT model with 12 layers and 12 attention heads for all BERT-based models. The same pretrained model checkpoint is used to provide a fair comparison and to observe the architecture that best benefits from the pretrained language model. Each classifier is trained using the Adam optimizer with a learning rate of 2e-5 and a dropout rate of 0.1.

4.2 Results and Analysis

In Table 3, we compare the performance of IA-BERT with several baseline models on Twitter and Reddit dataset. The proposed model outperforms the baselines in both online discourse datasets, Twitter and Reddit. It yields a performance improvement from the standard BERT classifier. LCF-BERT [27] shows the most competitive performance among the baselines. The model also uses pretrained BERT embedding to obtain each token representations. The experiment result shows that IA-BERT outperforms the LCF-BERT and gains the most benefit from the pretrained language model compared to the other BERT-based architectures.

The performance of IA-BERT on the number of last contexts is shown in Table 4. The proposed architecture leverages both the response and context to predict whether the response sequence is sarcastic or not. For the Twitter dataset, IA-BERT obtains the best performance when the response and the last two utterances are used as input which results in 0.805 accuracy. In the Reddit dataset, it is achieved when the model takes the last three utterances of the context sequence. In this setting, IA-BERT obtains 0.700 accuracy. Considering a number of utterances happened before the response sequence can give enough relevant information about the topic under discussion. Taking the last two or three utterances appears to be effective in identifying sarcasm while referring to the last context only or more than three can be either insufficient or less informative.

To study the importance of each attention module in IA-BERT, we conduct an ablation study by evaluating the model performance when removing a certain layer or module. The results of the ablation study on the Twitter and Reddit dataset can be seen in Table 5. Removing the feature attention layer and directly concatenating the output from the response attention and context-response attention layer constantly drops the prediction accuracy. It shows that the incongruity attention layer plays an important role in improving the model performance.

In the Twitter dataset, the prediction accuracy and F1-score when considering only the feature from response attention (RA) or context-response attention (CRA) layer are quite similar. Solely concatenating features from the two attention modules degrades the model

Context	“Hi . Just woke up . Witcher marathon last night . Worth it .” “@USER I did the same the other day and I ’ m ready for the next season”	“So Trump got Ukraine to announce an investigation after all — against him . <URL>” “@USER @USER Disgusting that Ukraine announces this investigation before DOJ and State Department . The Trump Administration is a disgrace .”
Response	“@USER It was too good . @USER Got really in to it as well ! Makes me want to go back and play from the first one , which I have on disc . That’s right . I had the game BEFORE it got really popular . #trendsetter”	“@USER @USER Not as Much as the Bidens and the Rest of the " Democratic " Party . #GOP #PartyOfLincoln #Democrats”
BERT	NOT_SARCASM	SARCASM
BERT-AEN	NOT_SARCASM	SARCASM
LCF-BERT	NOT_SARCASM	SARCASM
IA-BERT	SARCASM (Correct)	NOT_SARCASM (Correct)
Label	SARCASM	NOT_SARCASM

(a) Twitter

Context	“Round 1 - Pick 13: Laremy Tunsil, OT, Ole Miss (Miami Dolphins)” “Why wont ESPN acknowledge that its a video and not a picture?”	“In an ominous report, researchers warn that as many as 19 various ’tipping points’ could be triggered by the increasingly warm temperatures in the world’s northern polar region.” “Your kids and grandkids are going to be so pissed.” “if you’re younger than 30, **you** will be so pissed.”
Response	“Can’t watch a video of the devil magic.”	“am 16, I am already absolutely livid”
BERT	NOT_SARCASM	SARCASM
BERT-AEN	NOT_SARCASM	SARCASM
LCF-BERT	NOT_SARCASM	SARCASM
IA-BERT	SARCASM (Correct)	NOT_SARCASM (Correct)
Label	SARCASM	NOT_SARCASM

(b) Reddit

Table 6: Result samples of Twitter and Reddit dataset. The left side is that the response sounds sarcastic after referring to the context. The right side is that the response sounds sarcastic but actually non sarcastic considering the context. The baseline models misclassified samples because they do not reflect context, but IA-BERT reflects the context-aware information using context-response feature.

accuracy but improves the F1 score. Applying the incongruity attention to these features has improved the performance in both accracy and F1 score. This results show that the incongruity attention module play an important role in identifying sarcasm. The same phenomenon is also observed in the Reddit dataset. While leveraging the features obtained from response attention (RA) and context-response attention (CRA) is competitive, the incongruity attention layer helps in boosting the performance.

Overall, the complete architecture of IA-BERT demonstrates a better performance than the baseline models. It also outperforms the sophisticated architecture of LCF-BERT [27].

4.3 Qualitative Analysis

While maintaining a high precision, IA-BERT constantly obtains the highest recall. It shows that IA-BERT has a better sensitivity to predict the sarcastic responses. In the real application, it is important to capture the intended sarcasm, for example in social media platforms. Sarcasm especially offensive messages are against the social norms and developers have been making an effort to automatically detect such messages and prevent them from harming the online society. As social media has widely used as a tool to collect public opinions, sarcasm becomes a negative impact since its implicit meaning may flip the actual sentiment. Therefore, identifying sarcasm is beneficial for opinion mining.

In this case, IA-BERT shows that it has a better ability in recognizing sarcastic responses compared to the competitive BERT-based baseline models. As shown in Table 6, the baseline models miss classify the sarcastic sample in Twitter and Reddit dataset. The response in both samples does not sound sarcastic when the context is not considered. For example, if we simply read the sentence, "Can’t watch a video of the devil magic." in Reddit sample, we may think that ’the devil magic’ is only a video title or a name of a character from a movie. But after reading the context sequence, we will notice that the phrase is a sarcastic way to refer a subject being discussed. A sarcastic expression can also be found in the Twitter sample where the sarcasm is used for humorous purpose. The response implies that the community is late in hyping the subject being discussed in the context. For these type of samples, IA-BERT correctly predicts the label by considering the context information.

In the case of predicting non sarcastic samples, IA-BERT also shows a good performance by considering the context sequences. In Table 6, baseline models predict the non sarcastic samples as sarcastic. These models may see the response as a sarcastic because of the offensive expression used in the response and fail to capture the relation with its context. The response samples of Twitter and Reddit are actually explicit opinions and not an intended sarcasm.

5 RELATED WORK

In our experiments, we compare the proposed model to previous works which are closely related to our work and originally evaluated on the same dataset. This section provides a brief description of each approach. On the shared task of sarcasm detection, Baruah et al. [2] attempt to build the sarcasm detection model using pretrained language model by fine-tuning BERT as a classifier. Dadu and Pant [4] propose a special context separator token to better fine tune the pretrained models for context-aware sarcasm detection. The special token creates a boundary between the response sequence and the context sequence.

Shangipour ataei et al. [23] leverage an interactive attention network to model the interaction between a response text in online discourse and its context. They initialize all word embeddings in the context and response sequences using the GloVe embedding model. They employ an LSTM network to obtain intermediate representations of each token and an attention mechanism to extract interactive features between response and context. Ma et al. [14] originally propose the interactive attention network to perform the targeted aspect-based sentiment analysis. Shangipour ataei et al. [23] modify the model for context-aware sarcasm detection by treating the response text as the target sequence.

An attentional encoder network, BERT-AEN [24] is also adapted to solve the sarcasm detection problem. The model takes the response sequences as target and the context sequences as context input [23]. BERT-AEN is built with an attentive encoder which consists of multi-head attention layer and point-wise convolution transformation layer [24]. BERT-AEN employs a self-attention over the context sequence representations while our proposed architecture uses the self-attention mechanism to extract features from the response sequence.

Shangipour ataei et al. [23] also alter the input for a context-based classifier, LCF-BERT, to perform the sarcasm prediction. LCF-BERT leverages multi-head self-attention and employs dynamic mask and dynamic weighted layers along with the BERT-shared layer to extract local and global context features [27]. All the architectures we have described above were previously evaluated on the related context-aware sarcasm detection dataset and most of them leverage a pretrained language model as a token embedding layer.

6 CONCLUSION

In this work, we propose IA-BERT, a novel classifier architecture to solve the context-aware sarcasm detection problem. The model emphasizes the importance of context sequences as much as the response itself to identify sarcasm. IA-BERT consists of BERT embedding layer and three attention modules to extract relevant incongruity features for sarcasm detection. This approach yields a performance improvement over the standard BERT classifier and outperforms previous works on two online discourse datasets, Twitter and Reddit.

ACKNOWLEDGMENTS

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00421, Artificial Intelligence Graduate School Program (Sungkyunkwan University)). This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ICT Creative Consilience program(IITP-2021-2020-0-01821) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation)

REFERENCES

[1] Francesco Barbieri and Horacio Saggion. 2014. Modelling Irony in Twitter: Feature Analysis and Evaluation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland, 4258–4264.

[2] 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*. Association for Computational Linguistics, Online, 83–87.

[3] Arman Cohan, Iz Beltagy, Daniel King, Bhavana Dalvi, and Dan Weld. 2019. Pretrained Language Models for Sequential Sentence Classification. 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)*. Association for Computational Linguistics, Hong Kong, China, 3693–3699.

[4] Tanvi Dadu and Kartikey Pant. 2020. Sarcasm Detection using Context Separators in Online Discourse. In *Proceedings of the Second Workshop on Figurative Language Processing*. Association for Computational Linguistics, Online, 51–55.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186.

[6] 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)*. European Language Resources Association (ELRA), Istanbul, Turkey, 392–398.

[7] 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*. Association for Computational Linguistics, San Diego, California, 161–169.

[8] Debanjan Ghosh, Avijit Vajpayee, and Muresan Smaranda. 2020. A Report on the 2020 Sarcasm Detection Shared Task. In *Proceedings of the Second Workshop on Figurative Language Processing*. Association for Computational Linguistics, Online, 1–11.

[9] Roberto González-Ibáñez, 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*. Association for Computational Linguistics, Portland, Oregon, USA, 581–586.

[10] Aditya Joshi, Pushpak Bhattacharyya, and Mark J. Carman. 2017. Automatic Sarcasm Detection: A Survey. *ACM Comput. Surv.* 50, 5, Article 73 (Sept. 2017), 22 pages.

[11] Aditya Joshi, Vaibhav Tripathi, Kevin Patel, Pushpak Bhattacharyya, and Mark Carman. 2016. Are Word Embedding-based Features Useful for Sarcasm Detection?. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, 1006–1011.

[12] Neel Kant, Raul Puri, Nikolai Yakovenko, and Bryan Catanzaro. 2018. Practical text classification with large pre-trained language models. *arXiv preprint arXiv:1812.01207* (2018).

[13] Akshay Khatri and Pranav P. 2020. Sarcasm Detection in Tweets with BERT and GloVe Embeddings. In *Proceedings of the Second Workshop on Figurative Language Processing*. Association for Computational Linguistics, Online, 56–60.

[14] Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive Attention Networks for Aspect-Level Sentiment Classification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 4068–4074.

[15] Paul K. Mandal and Rakeshkumar Mahto. 2019. Deep CNN-LSTM with Word Embeddings for News Headline Sarcasm Detection. In *16th International Conference on Information Technology-New Generations (ITNG 2019)*, Shahram Latifi (Ed.). Springer International Publishing, Cham, 495–498.

[16] Bo Pang and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.* 2, 1–2 (Jan. 2008), 1–135.

[17] Soujanya Poria, Erik Cambria, Devamanyu Hazarika, and Prateek Vij. 2016. 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*. The COLING 2016 Organizing Committee, Osaka, Japan, 1601–1612.

[18] 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 (2020), 17309–17320.

[19] 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*. Dublin City University and Association for Computational Linguistics, Dublin, Ireland, 213–223.

[20] Antonio Reyes, Paolo Rosso, and Davide Buscaldi. 2012. From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering* 74 (2012), 1–12. Applications of Natural Language to Information Systems.

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

[22] 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*. Association for Computational Linguistics, Seattle, Washington, USA, 704–714.

[23] Taha Shangipour ataei, Soroush Javdan, and Behrouz Minaei-Bidgoli. 2020. Applying Transformers and Aspect-based Sentiment Analysis approaches on Sarcasm Detection. In *Proceedings of the Second Workshop on Figurative Language Processing*. Association for Computational Linguistics, Online, 67–71.

[24] Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314* (2019).

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NIPS*. 6000–6010.

[26] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. 2017. Multi-Modal Factorized Bilinear Pooling With Co-Attention Learning for Visual Question Answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[27] Biqing Zeng, Heng Yang, Ruyang Xu, Wu Zhou, and Xuli Han. 2019. LCF: A Local Context Focus Mechanism for Aspect-Based Sentiment Classification. *Applied Sciences* 9, 16 (2019).