International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-021-01344-2

ORIGINAL ARTICLE

Learning to capture contrast in sarcasm with contextual dual-view attention network

Lu Ren[1] · **Hongfei Lin**[1] · **Bo Xu**[1] · **Liang Yang**[1] · **Dongyu Zhang**[1]

Received: 10 October 2020 / Accepted: 12 May 2021

Abstract
Sarcasm is a common way of rhetoric in our daily life. It is used to express the opposite of the literal meaning, which makes it a challenging task in sentiment analysis of natural language processing (NLP). The formation mechanism of sarcasm is usually caused by the contrast between the positive sentiment and the negative situation. In this paper, we propose a contextual dual-view attention network (CDVaN) for sarcasm detection according to the formation mechanism of sarcasm. A Contrast Understanding Unit is proposed to effectively extract the contrast between the positive sentiment and the negative situation from the view of formation mechanism of sarcasm. Apart from it, we further use a Context Understanding Unit to extract the contextual semantic information from the contextual semantic view. Our experiments on the IAC-V1 dataset and IAC-V2 dataset demonstrate that the proposed CDVaN model can distinguish sarcasm effectively. The results show that our model achieves state-of-the-art or comparable results.

Keywords Sarcasm detection · Sentiment analysis · Natural language processing · Dual-view attention network

1 Introduction

Sarcasm is defined in the Free Dictionary[1] as a cutting, often ironic remark, which is used to express contempt or ridicule. Sarcasm is used to express our sentiment subtlety in our daily life, such as dislike, dissatisfaction and so on. Gibbs [11] presented that about 8% of the conversations among friends were sarcastic expressions. The surficial sentiment of sarcasm maybe positive, but it usually expresses strong negative sentiment. Due to the opposite polarity between the literal sentiment and the actual sentiment, the study of sarcasm can promote the development of sentiment analysis in natural language processing (NLP). The task of sarcasm detection refers to determining whether a sentence in a given context expresses a certain degree of sarcasm. Namely, sarcasm detection is to judge the sentence is sarcasm or not. Sarcasm detection attracts more and more attention of the NLP researchers in recent years and can be used in many NLP applications, including sentiment analysis, human-machine dialogue and so on. Existing sarcasm detection algorithms can be divided into several types, including text-based sarcasm detection, multi-modal sarcasm detection and so on. In this paper, we only focus on text-based sarcasm detection.

Sarcasm is a kind of verbal behavior, which is used as the form of positive or affirmative words to express the negative evaluation [34]. A sarcastic sentence is usually caused by the contrast between positive sentiment and negative situation [34]. There are two different kinds of negative situations, including (1) the situation with direct negative sentiment, and (2) the ordinary situation which becomes negative situation in specific sentence. To better illustrate the characteristics of sarcasm, we provide some examples as follows.

As shown in Table 1, Exp.1 is the first case of sarcasm, which is caused by the contrast between positive sentiment word ("love") and negative situation with direct negative

✉ Lu Ren
renlu@mail.dlut.edu.cn

✉ Hongfei Lin
hflin@dlut.edu.cn

Bo Xu
xubo@dlut.edu.cn

Liang Yang
liang@dlut.edu.cn

Dongyu Zhang
zhangdongyu@dlut.edu.cn

1 Dalian University of Technology, Dalian, China

[1] https://www.thefreedictionary.com/.

Table 1 Examples of sarcasm or not

Number	Text	Sarcasm or not
Exp.1	I love to be woken up by the noise	Sarcasm
Exp.2	The movie is so wonderful that I fall asleep	Sarcasm
Exp.3	I hate to be woken up by the noise	Non sarcasm

sentiment ("noise"). Since no one loves to be woken up by noise in real life, it is sarcasm. Strong dissatisfaction is expressed by positive sentiment word ("love") in the sentence.

Exp.2 is the second case of sarcasm, which is caused by the contrast between the positive sentiment word ("wonderful") and the ordinary situations ("movie" and "fall asleep"). Generally speaking, "movie" and "fall asleep" are ordinary situations. But when you say "the movie makes you fall asleep", it's a negative situation. Sarcasm is generated by the use of positive sentiment ("wonderful") to describe the negative situation ("movie" makes you "fall asleep").

The expression of Exp.3 is not sarcasm, which contains the negative sentiment word ("hate") and the negative situation ("noise"). The negative sentiment has no contrast with the the negative situation. So, Exp.3 is not sarcasm. And it can prove that the polarity of sentiment is very important in sarcasm detection.

As shown from the examples, the polarity of sentiment is very important in sarcasm. In order to distinguish sarcasm effectively, we need to distinguish the polarity of the sentiment expressed in the sentence. That is to say, we need to capture the contrast between positive sentiment and negative situations in sarcasm detection.

Early research on sarcasm detection relied on manual extraction of features, including synonyms [4], semantic relatedness [33], sentiments [12] and so on. Since feature extraction involves a lot of manual labor and lacks of rich semantic information contained in the text, many researches have tried to use deep learning to detect sarcasm in recent years [31, 36]. Tay et al. [36] proposed the multi-dimensional intra-attention recurrent network to capture the contrast and incongruity. They did not take into consideration of the sentiment information in sarcasm. Ren et al. [31] proposed a multi-level memory network based on sentiment semantics to capture the contrast between sentiment semantics and words in the sentence. They did not take into consideration of the polarity of sentiment in sarcasm detection.

In order to capture the contrast in sarcasm, we present a contextual dual-view attention network (CDVaN), which extracts the features of sarcasm from two different views with different units simultaneously, including a Context Understanding Unit and a Contrast Understanding Unit. From the view of formation mechanism of sarcasm, We divide the text into positive sentiment part and negative situation part with SenticNet. We capture the contrast between positive sentiment and negative situation in text though a multi-hop attention network in Contrast Understanding Unit. From the view of contextual semantic view, we further apply a multi-hop attention network based on Long Short Term Memory (LSTM) to capture contextual semantic information in Context Understanding Unit.

The contributions of this paper can be summarized as follows:

- We present a contextual dual-view attention network to capture contextual semantic information and the contrast between positive sentiment and negative situation in sarcasm. We apply a multi-hop attention network to capture the contrast from the view of formation mechanism of sarcasm. We also capture the contextual semantic information through a multi-hop attention network from the contextual semantic view.
- We divide the text into two semantic parts with Senticnet, including the positive sentiment part and the negative situation part. Meanwhile, a Contrast Understanding Unit is proposed to extract the contrast between the positive sentiment part and the negative sentiment part. Ablation studies verify the effectiveness.
- We conduct experiments on IAV-V1 dataset and IAC-V2 dataset. The experimental results indicate our model can get state-of-the-art or comparable performance.

The structure of this paper is organized as follows. Section 2 gives the related work of sarcasm detection. Section 3 presents the details of our model. Section 4 shows our experimental results. Section 5 gives conclusion and future work.

2 Related work

Sarcasm is a form of expressions which used to express the opposite of the literal meaning. The purpose of sarcasm detection is to judge the text is sarcasm or not. In this section, we provide a brief review of methods which are closed to our work, including sarcasm detection and attention mechanism in NLP.

Sarcasm detection can improve the development of sentiment analysis in NLP research. Sarcasm detection has attracted the attention of many scholars. In this subsection, three types of sarcasm detection algorithms are introduced, including feature-engineering based methods, deep learning methods and multi-modal sarcasm detection methods. We review these three types of sarcasm detection methods briefly.

2.1 Feature-engineering based method

Early researches on sarcasm detection were based on feature engineering method. Many salient feature sets were proposed, including N-grams [21, 32], ambiguity-based [4, 33] sentiment-based [3, 17, 29] and others. These features were useful for sarcasm detection.

Maynard and Greenwood [24] used the hashtag sentiment for sarcasm detection. Bharti et al. [5] proposed two models for sarcasm detection based on parsing-based lexicon and interjection words. Farías et al. [8] proposed a sarcasm detection method based on semantic resources, including SentiWordnet, AFINN and General Inquirer. References [17, 18] proposed sarcasm detection methods based on harnesses context incongruity and semantic similarity or discordance between words. Joshi et al. [16] presented sarcasm detection and sentiment classification interact and have the strong relationship. Riloff et al. [34] proposed a sarcasm detection model that can select phrases which contain positive sentiment and negative situation from tweets.

2.2 Deep learning method

With the development of deep learning in NLP, more and more deep learning based sarcasm detection methods are proposed. Ghosh and Veale [9] proposed a deep neural network based on CNN and LSTM for sarcasm detection. Ghosh and Veale [10] proposed a neural network for sarcasm detection based on CNN and LSTM which considered psychological dimensions of speakers. Tay et al. [36] proposed an attention-based neural network named Multi-dimensional Intra-Attention Recurrent Network to detect sarcasm. Ren et al. [30] proposed a convolutional neural networks based on context-augmented for sarcasm detection. Ren et al. [31] proposed a multi-level memory network using sentiment semantics to capture the features of sarcasm expressions. The information of user was considered in sarcasm detection [1, 14, 19].

2.3 Multi-modal sarcasm detection

In recent years, many researchers have begun multi-modal studies of sarcasm detection. Mishra et al. [26] proposed a sarcasm detection based on human cognition. They considered eye movements when people read the text. Mishra et al. [25] proposed a convolutional neural network based on human cognitive for sarcasm detection. Cai et al. [6] proposed a hierarchical fusion model to fusion multi-modal features, including text, image and attribute. Castro et al. [7] proposed a new Multimodal sarcasm dataset and a sarcasm detection method based on text features, speech features and video features.

2.4 Attention mechanism in NLP

Attention mechanism has been successfully used in many NLP tasks, such as machine translation [2, 23], question answering [20, 37], text classification [39] and sentiment analysis[35]. The intuition of attention mechanism is that the representation in each low-level position owns different importance for the high-level representation. Moreover, attention mechanism is also used in sarcasm detection task [31, 36]. Overall, we apply the attention mechanism to capture the contrast in sarcasm in this paper.

3 Methods

In this section, we give a brief introduction of the proposed contextual dual-view attention network(CDVaN) for sarcasm detection. Our model extracts the features of the sarcasm from two different views, including the formation mechanism view and the contextual semantic view. The Contrast Understanding Unit and the Context Understanding Unit are proposed to extract the features from these two views respectively. The Contrast Understanding Unit is used to capture the features, which encodes the contrast between the positive sentiment and negative situation from the view of formation mechanism of sarcasm. The Context Understanding Unit is used to capture the contextual information from the contextual semantic view. The overall architecture of our model is shown in Fig. 1. In the following, we describe the details of the two units in CDVaN model.

3.1 Context Understanding Unit

The Context Understanding Unit contains three layers, including an input layer, a contextual encoding layer and a multi-hop attention layer, which can extract the contextual features hierarchically. The details of these three layers are presented as follows.

Input layer In Context Understanding Unit, the input is the whole text. Input layer maps each word into low-dimensional vector space which contains semantic information. Here, we apply the pre-trained word vectors, GloVe [28] to get the word embedding for each word in this part. The dimension of word embedding is represented as d.

Contextual encoding layer Long Short Term Memory (LSTM) was proposed by Hochreiter and Schmidhuber [15] and has been widely used in NLP research. But the LSTM only considers the text information before the current word. Based on LSTM, [13] presented bidirectional LTSM, called Bi-LSTM, which can capture the contextual

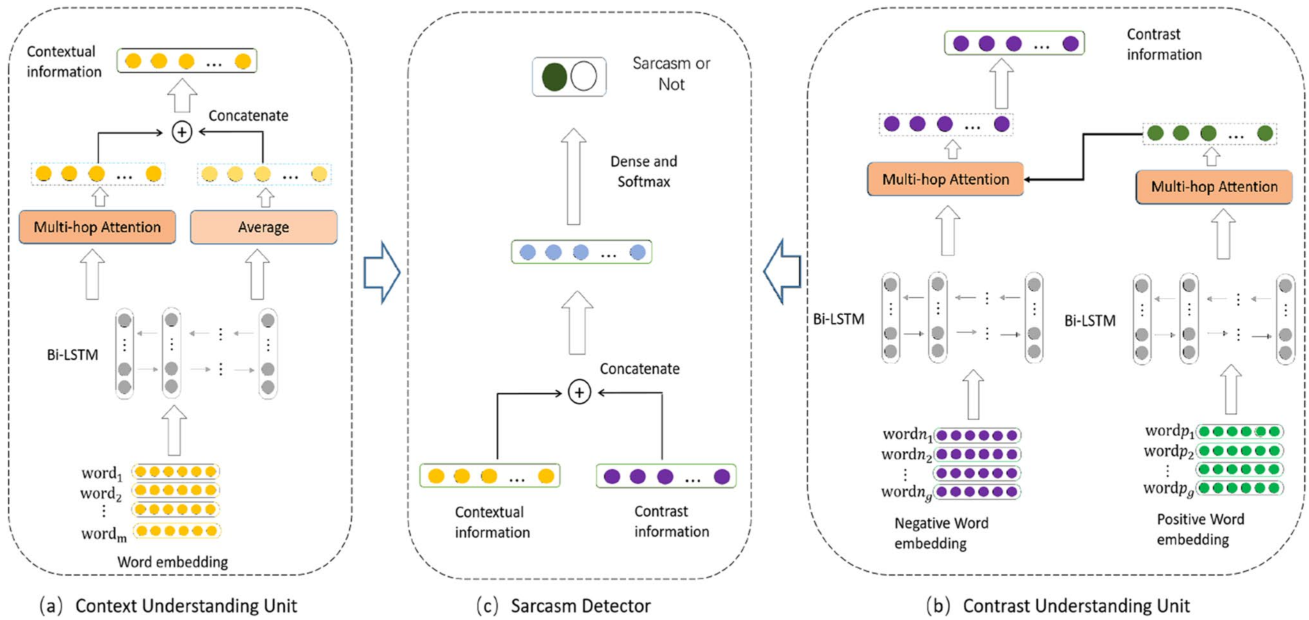

Fig. 1 The structure of CDVaN model. There are two units in CDVaN model, including a Context Understanding Unit and a Contrast Understanding Unit

information from both directions. Bi-LSTM contains a forward LSTM and a backward LSTM. Due to the advantage of Bi-LSTM, we apply Bi-LTSM to capture the contextual information in Context Understanding Unit. The forward and backward LSTMs both own three gates (including an input gate i_t, a forget gate f_t and an output gate o_t) and a memory cell c_t. The operations of LSTM are shown as the following formulas.

$$i_t = \sigma\left(W_i x_i + U_i h_{t-1}\right) \tag{1}$$

$$f_t = \sigma\left(W_f x_i + U_f h_{t-1}\right) \tag{2}$$

$$o_t = \sigma\left(W_o x_i + U_o h_{t-1}\right) \tag{3}$$

$$\tilde{c}_t = \tanh\left(W_c x_i + U_c h_{t-1}\right) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c} \tag{5}$$

$$h_t = o_t \odot \tanh\left(c_t\right) \tag{6}$$

where x_i is the current input word vector, σ denotes the sigmoid function and $\odot$ means the element-wise multiplication operation. W_i, W_f, W_o, W_c, U_i, U_f, U_o and U_c are the weight matrix. h_t is the hidden state vector. Due to LSTM only considering the forward information in the sentence, we apply Bi-LSTM to obtain the contextual information.

$$h_{Bilstm} = concatenate[\vec{h};\overleftarrow{h}] \tag{7}$$

$\vec{h}$ denotes the output of forward LSTM and $\overleftarrow{h}$ denotes the output of backward LSTM. The output matrix of Bi-LSTM can be expressed as $F_{context} = \{h_1, h_2, \dots, h_n\}$.

Multi-hop attention layer The goal of attention mechanism is to assign different weights with different parts. In the field of NLP, attention mechanism is usually used to capture the import words in the sentence for specific task. According to that multi-hop attention network could capture the complex semantic relations [38], we apply multi-hop attention mechanism in our model. $F_{context}$ is the input of this part. The attention mechanism is represented with the following formulas.

$$h_i' = \tanh\left(W_c h_i + b_c\right) \tag{8}$$

$$a_i = \frac{\exp\left({h'}_i^T q_c\right)}{\sum_{j=1}^{n} \exp\left({h'}_i^T q_c\right)} \tag{9}$$

$$q_{new} = \sum_{i=1}^{n} a_i h_i \tag{10}$$

where W_c and b_c denote the weight matrix. h_i is the new representation of the ith word after encoding by Bi-LSTM. q_c

denotes a high-level representation of the query "what is the important word in the text for sarcasm detection". q_c is randomly initialized and can be changed in training processing. q_{new} denotes the output of one hop attention layer. q_c of next hop can be undated by the formula $q_c = q_c + q_{new}$. The output q_{new} in last hop is the output of multi-hop attention layer.

Finally, the output of the Context Understanding Unit can be achieved by the follow operations.

$$f_{average} = \frac{\sum_{i=1}^{n} h_i}{n} \tag{11}$$

$$H_{Context} = concatenate[f_{average};q_{new}] \tag{12}$$

where $f_{average}$ denotes the average value of matrix $F_{context}$. $H_{Context}$ denotes the output of Context Understanding Unit.

3.2 Contrast Understanding Unit

Due to the formation mechanism of sarcasm, we design the Contrast Understanding Unit. Sarcasm is caused by the contrast between positive sentiment and negative situation. The Contrast Understanding Unit contains three layers, including an input layer, a positive sentiment understanding layer and a contrast understanding layer. The details of these three layers are presented as follows.

Input layer The words in the text are divided into two semantic parts with Senticnet[2], including positive sentiment part and the negative situation part. The words labeled as positive sentiment are assigned into the positive sentiment part, while the neutral words and the negative sentiment words are assigned into the negative situation part. The neutral words are assigned into negative situation part, because negative situation may consists of ordinary situations, as shown in Exp.2 of Table 1. For example, the sentence "I love toothache" is divided to positive sentiment part "love" and the negative situation part "I toothache". We also map these two semantic parts into the matrix of word vectors, as in Sect. 3.1, named X_{pos} and X_{neg} respectively.

Positive sentiment understanding layer The goal of positive sentiment understanding layer is to obtain positive emotional semantic information in the text. We also apply multi-hop attention layer based on Bi-LSTM in this layer, as Sect. 3.1. The output of last hop attention layer q_{pos} is as the output of positive sentiment understanding layer.

Contrast Understanding Layer The inputs of Contrast Understanding Layer are X_{neg} and q_{pos}. There are two steps in Contrast Understanding Layer. We also apply Bi-LSTM to capture negative situation information h_{neg} with the input X_{neg}, where $h_{neg} = \{H_1, H_2, \ldots, H_n\}$. Tran and Niedereée [38] proposed that multi-hop attention network could capture the complex semantic relations. Inspired of this, we also apply multi-hop attention layer to capture the contrast between positive sentiment and negative situation in text. h_{neg} and q_{pos} are the inputs of attention mechanism. The formulas are as following.

$$H'_i = \tanh\left(W_n H_i + b_n\right) \tag{13}$$

$$A_i = \frac{\exp\left({H'}_i^T q_{pos}\right)}{\sum_{j=1}^{n} \exp\left({H'}_i^T q_{pos}\right)} \tag{14}$$

$$q_{contrast} = \sum_{i=1}^{n} A_i H_i \tag{15}$$

where W_n and b_n denote the weight matrix. H_i is the new representation of the ith word in X_{neg} after encoding by Bi-LSTM. q_{pos} denotes a high-level representation of positive sentiment information in the text for sarcasm detection. $q_{contrast}$ denotes the output in this hop attention mechanism. q_{pos} in next hop can be undated by the formula $q_{pos} = q_{pos} + q_{contrast}$. The output $q_{contrast}$ in last hop is the output of multi-hop attention layer.

3.3 Loss function

As described above, we obtain two semantic features from the two units, including contextual information $H_{Context}$ and contrast information $q_{contrast}$. We concatenate these features to obtain the final representation f_{final}:

$$f_{final} = concatenate[H_{Context};q_{contrast}] \tag{16}$$

Accordingly, we apply softmax function in output layer. The formula of softmax function is as following:

$$y = soft\max\left(W_h f_{final} + b_h\right) \tag{17}$$

We apply cross-entropy loss for sarcasm detection in our model. Our model is trained by minimizing the cross-entropy loss. The formula is as following.

$$loss = -\sum_i \sum_j y_i^j \log \hat{y}_i^j \tag{18}$$

[2] https://www.sentic.net/downloads/.

Table 2 Statistics of internet argument corpus (IAC-V1 and IAC-V2) dataset

Dataset	Train	Dev	Test
IAC-V1	1549	193	193
IAC-V2	3762	464	466

We apply the standard measures precision (P), recall (R) and macro-averaged F1-score (F1) to evaluate the effectiveness for sarcasm detection. Macro-averaged F1-score implies that all class labels have equal weight in the final score.

Table 3 Comparison of different models of sarcasm detection (%)

Model	IAC-V1			IAC-V2		
	P	R	F1	P	R	F1
CNN[a]	58.21	58.00	57.95	68.45	68.18	68.21
LSTM[a]	54.87	54.89	54.84	68.30	63.96	60.78
Attention LSTM[a]	58.98	57.93	57.23	70.04	69.62	69.63
GRNN[a] [40]	56.21	56.21	55.96	62.26	61.87	61.21
CNN-LSTM-DNN[a] [9]	55.50	54.60	53.31	64.31	64.33	64.31
MIARN[a] [36]	63.88	63.71	63.18	72.92	72.93	72.75
MMNSS [a] [31]	66.86	**70.93**	67.67	**75.00**	71.05	74.20
CDVaN	**67.01**	69.15	**68.39**	74.80	**77.31**	**75.07**

The significances of bold in the table denote the best results

[a]Means that the results are shown in other literatures

where y denotes the target distribution and $\hat{y}$ denotes the predicted distribution. i is the index of text, and j is the index of class.

4 Experiments and results

4.1 Datasets

Experiments are conducted on the two publicly available benchmark datasets to detect sarcasm which are collected from Internet Argument Corpus (IAC) [27, 27][3]. The IAC corpus is designed to study political debates on BBS online. There are two versions of IAC datasets, called IAC-V1 [27] and IAC-V2 [27]. And the details of datasets are shown in Table 2.

4.2 Implementation details and metrics

We apply the pre-trained GloVe embedding[28] and the dimension is 300 in our expriments. In Context Understanding Unit, the unit size of LSTM is 32 and the hop in multi-hop attention layer is 3. In Contrast Understanding Unit, the unit size of LSTM is 32 and the hop in multi-hop attention layer is 3. The dropout in our model is 0.4. The mini-batch size is 64.

4.3 Comparison with existing methods

In this subsection, we verify the effectiveness of our model by comparing with many state of the art methods, including CNN, LSTM, Attention based on LSTM, GRNN [40], CNN-LSTM-DNN [9], MIARN [36] and MMNSS [31].

- **CNN.** CNN is the vanilla Convolutional Neural Network used in NLP. CNN can capture the n-gram information in the sentence. We use the results shown in [36]. The filter size in CNN is 3 and the filter number is 100.
- **LSTM** LSTM is the vanilla Long Short-Term Memory Network with the unit size as 100. We also use the results shown in [36].
- **ATT-LSTM** ATT-LSTM stands for attention based on LSTM. We use the results shown in Tay et al. [36]. Tay et al. [36] follow [39] and use the document-level attention network.
- **GRNN** GRNN stands for Gated Recurrent Neural Network. GRNN is proposed by Zhang et al. [40] to detect sarcasm in tweet. It is a Bidirectional Gated Recurrent Unit model. We use the results shown in Tay et al. [36].
- **CNN-LSTM-DNN** CNN-LSTM-DNN is a model that combining CNN, LSTM and Deep Neural Network(DNN) for sarcasm detection. It is proposed by Ghosh and Veale [9] and contains 2 layer CNN, 2 layer LSTM and DNN layer. We use the results shown in Tay et al. [36].
- **MIARN** MIARN is proposed by Tay et al. [36]. MIARN model applies a multi-dimensional intra-attention based on LSTM for sarcasm detection.

[3] https://nlds.soe.ucsc.edu/sarcasm1.

Table 4 Comparison of Different Units in CDVaN model on IAV-V1 dataset (%)

	P	R	F1
CeUU	**68.63**	67.31	65.65
CaUU	66.67	**71.29**	66.09
CDVaN	67.01	69.15	**68.39**

The significances of bold in the table denote the best results

Table 5 Comparison of different contrasts in CDVaN model on IAV-V1 dataset (%)

	P	R	F1
PT in CDVaN	**72.60**	53.54	65.39
PN in CDVaN	66.32	65.63	66.32
CDVaN	67.01	**69.15**	**68.39**

The significances of bold in the table denote the best results

– **MMNSS** MMNSS is a model for sarcasm detection which stands for a multi-level memory network based on sentiment semantics [31].

The experimental results are shown in Table 3. CDVaN model is verified the performance on IAC-V1 dataset and IAC-V2 dataset. From the results shown in Table 3, we have several observations as following.

– Attention mechanism can get a better performance in sarcasm detection. The models based on attention mechanism (Attention LSTM, MIARN, MMNSS and CDVaN) are better or comparable than the models without attention mechanism. Because attention mechanism can capture the important words in sentence for sarcasm detection.
– The performance of CDVaN model is better than MIARN model. MIARN model considers to capture the contrast in sarcasm with the attention mechanism based on word pairs. It is not enough to capture the contrast in sarcasm with word pairs.
– The results on CDVaN model are better than MMNSS model. MMNSS applies multi-level memory network to capture the contrast in sarcasm. MMNSS model considers to capture the contrast between all sentiment words and all text. It may introduce more noises. And MMNSS model doesn't consider the polarity of sentiment, so the performance of our CDVaN model is better than MMNSS model.

4.4 Detailed analysis

In this subsection, we design a series of experiments on IAC-V1 dataset to prove the effectiveness of CDVaN model from two perspectives, including the impact of both units in CDVaN model and the impact of different contrasts in CDVaN model. We further analyze our model through error analysis.

The impact of both units in CDVaN model In order to verify the effectiveness of Context Understanding Unit(CeUU) and Contrast Understanding Unit(CaUU), we only use one unit to detect sarcasm. The results are shown in Table 4.

As shown in Table 4, Context Understanding Unit can bring 2.3 % improvement and Contrast Understanding Unit can bring 2.74 % improvement. It can demonstrate that both units are effective, and Contrast Understanding Unit is more important in our model.

The impact of different contrasts in CDVaN model In order to prove the effectiveness of the contrast between positive sentiment and negative situation in CDVaN, we design different contrasts in this subsection, including contrast between positive sentiment and all text (called PT in CDVaN), contrast between positive sentiment and negative sentiment (called PN in CDVaN). CDVaN applies the contrast by the positive words and other words in sentence. PT in CDVaN applies the contrast by the positive words and all words in sentence. PN in CDVaN applies the contrast by the positive sentiment and negative sentiment. The results on IAC-V1 dataset are shown in Table 5.

From the results shown in Table 5, PT in CDVaN, which considers the contrast between positive sentiment and all words in sentence, get the worst result. It is because that all words would introduce the more noises than other models. The results of PN in CDVaN are better. It's because that PN in CDVaN model consider the contrast between positive sentiment and negative sentiment. But PN in CDVaN doesn't consider the case shown as Exp.2 in Table 1 ("The movie is so wonderful that I fall asleep."). The case has no obvious negative sentiment information in the sentence. Due to considering the contrast that shown as Exp.2, CDVaN model gets the best results. CDVaN model not only considers the contrast of various cases, but also does not introduce too much noise.

Error analysis We further analyze our model through error analysis in this part. Examples of our model prediction errors are shown in Table 6.

From the examples 01 and 02 shown in Table 6, we can see the errors that are due to the fact that the sentences contain the contrast between positive and negative

Table 6 Error analysis of CDVaN model

Number	Text	Label	prediction label
01	A perfect example of why Christian fundamentalism and evolution can't co-exist. If God is wrong, then the Bible is wrong. Evolution contradicts the Bible, plain and simple	Non sarcasm	Sarcasm
02	There's a book my mom has called "The trouble with Islam" by some lady. I don't know her name, but it looks like it may bee a good book for you	Non sarcasm	Sarcasm
03	Just so you don't shoot them!	Sarcasm	Non sarcasm
04	Now, I could ask the same thing	Sarcasm	Non sarcasm

sentiment. In example 01, the word "wrong" and the word "perfect" cause the contrast. In example 02, the word "good" and the word "trouble" make up the contrast. From the examples 03 and 04 shown in Table 6, we can see the errors that are due to the lack of context, that is background information. In future work, we will import more background information for the sarcasm detection task, such as knowledge graph, multi-modal information and so on.

5 Conclusion and future work

In this paper, we consider the contrast between positive sentiment and negative situation in sarcasm. Based on the formation mechanism of sarcasm, we propose a contextual dual-view attention network, called CDVaN model, to capture the contextual semantic information and the contrast in the sentence from different views. The experimental results show that CDVaN model achieves the state-of-the-art or comparable results.

With the development of sarcasm detection research, more and more studies don't only focus on text. Different modal information (such as text, image, voice and so on) can bring more semantic information to the sarcasm detection task. In the future, we will continue the sarcasm study based on multi-modal information.

Acknowledgements This work is partially supported by grant from the Natural Science Foundation of China (Nos. 62076046, 61632011, 62006034, 61772103), the Fundamental Research Funds for the Central Universities, the Ministry of Education Humanities and Social Science Project (No. 19YJCZH199), the Foundation of State Key Laboratory of Cognitive Intelligence, iFLYTEK, P.R. China (COGOS-20190001, Intelligent Medical Question Answering based on User Profiling and Knowledge Graph).

Declaration

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Amir S, Wallace BC, Lyu H, Silva PCMJ (2016) Modelling context with user embeddings for sarcasm detection in social media. arXiv preprint arXiv:160700976
2. Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. arXiv: Computation and Language
3. Bamman D, Smith NA (2015) Contextualized sarcasm detection on twitter. In: Ninth International AAAI Conference on Web and Social Media
4. Barbieri F, Saggion H, Ronzano F (2014) Modelling sarcasm in twitter, a novel approach pp 50–58
5. Bharti SK, Babu KS, Jena SK (2015) Parsing-based sarcasm sentiment recognition in twitter data. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, ACM, pp 1373–1380
6. Cai Y, Cai H, Wan X (2019) Multi-modal sarcasm detection in twitter with hierarchical fusion model pp 2506–2515
7. Castro S, Hazarika D, Pérez-Rosas V, Zimmermann R, Mihalcea R, Poria S (2019) Towards multimodal sarcasm detection (an_ obviously_ perfect paper)
8. Farías DIH, Patti V, Rosso P (2016) Irony detection in twitter: the role of affective content. ACM Trans Internet Technol (TOIT) 16(3):19
9. Ghosh A, Veale T (2016) Fracking sarcasm using neural network. In: Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis, pp 161–169
10. Ghosh A, Veale T (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, pp 482–491
11. Gibbs RW (2000) Irony in talk among friends. Metaphor Symbol 15(1–2):5–27
12. Gonzalezibanez R, Muresan S, Wacholder N (2011) Identifying sarcasm in twitter: a closer look pp 581–586
13. Graves A, Jaitly N, Mohamed A (2013) Hybrid speech recognition with deep bidirectional lstm. In: 2013 IEEE workshop on automatic speech recognition and understanding, IEEE, pp 273–278
14. Hazarika D, Poria S, Gorantla S, Cambria E, Zimmermann R, Mihalcea R (2018) Cascade: contextual sarcasm detection in online discussion forums. arXiv preprint arXiv:180506413
15. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
16. Joshi A, Bhattacharyya P, Carman MJ (2017) Automatic sarcasm detection: a survey. ACM Comput Surveys (CSUR) 50(5):73
17. Joshi A, Sharma V, Bhattacharyya P (2015) Harnessing context incongruity for sarcasm detection. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics

and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), vol 2, pp 757–762
18. Joshi A, Tripathi V, Patel K, Bhattacharyya P, Carman M (2016) Are word embedding-based features useful for sarcasm detection? arXiv preprint arXiv:161000883
19. Kolchinski YA, Potts C (2018) Representing social media users for sarcasm detection. arXiv preprint arXiv:180808470
20. Kumar A, Irsoy O, Ondruska P, Iyyer M, Bradbury J, Gulrajani I, Zhong V, Paulus R, Socher R (2015) Ask me anything: Dynamic memory networks for natural language processing. arXiv: Computation and Language
21. Liebrecht C, Kunneman FA, Den Bosch AV (2013) The perfect solution for detecting sarcasm in tweets #not pp 29–37
22. Lukin SM, Walker MA (2017) Really? well. Apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. arXiv: Computation and Language
23. Luong MT, Pham H, Manning CD (2015) Effective approaches to attention-based neural machine translation. Comput Sci. arXiv:1508.04025
24. Maynard D, Greenwood MA (2014) Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis. In: LREC 2014 Proceedings, ELRA
25. Mishra A, Dey K, Bhattacharyya P (2017a) Learning cognitive features from gaze data for sentiment and sarcasm classification using convolutional neural network. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp 377–387
26. Mishra A, Kanojia D, Nagar S, Dey K, Bhattacharyya P (2017b) Harnessing cognitive features for sarcasm detection. arXiv preprint arXiv:170105574
27. Oraby S, Harrison V, Reed L, Hernandez E, Riloff E, Walker M (2017) Creating and characterizing a diverse corpus of sarcasm in dialogue. arXiv preprint arXiv:170905404
28. Pennington J, Socher R, Manning C (2014) Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
29. Rajadesingan A, Zafarani R, Liu H (2015) Sarcasm detection on twitter: a behavioral modeling approach. In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, ACM, pp 97–106
30. Ren Y, Ji D, Ren H (2018) Context-augmented convolutional neural networks for twitter sarcasm detection. Neurocomputing 308:1–7
31. Ren L, Xu B, Lin H, Liu X, Liang Y (2020) Sarcasm detection with sentiment semantics enhanced multi-level memory network. Neurocomputing 401:320–326
32. Reyes A, Rosso P (2012) Making objective decisions from subjective data: detecting irony in customer reviews. Decision Support Syst 53(4):754–760
33. Reyes A, Rosso P, Buscaldi D (2012) From humor recognition to irony detection: the figurative language of social media. Data Knowl Eng 74:1–12
34. Riloff E, Qadir A, Surve P, De Silva L, Gilbert N, Huang R (2013) Sarcasm as contrast between a positive sentiment and negative situation. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp 704–714
35. Tang D, Qin B, Liu T (2016) Aspect level sentiment classification with deep memory network pp 214–224
36. Tay Y, Tuan LA, Hui SC, Su J (2018) Reasoning with sarcasm by reading in-between. arXiv preprint arXiv:180502856
37. Tran NK, Niedereee C (2018) Multihop attention networks for question answer matching pp 325–334
38. Tran NK, Niedereée C (2018) Multihop attention networks for question answer matching. pp 325–334, https://doi.org/10.1145/3209978.3210009
39. Yang Z, Yang D, Dyer C, He X, Smola A, Hovy E (2016) Hierarchical attention networks for document classification. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies
40. Zhang M, Zhang Y, Fu G (2016) Tweet sarcasm detection using deep neural network. In: Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers, pp 2449–2460