Augmenting Data for Sarcasm Detection with Unlabeled Conversation Context

Hankyol Lee
RippleAI
Seoul, Korea
hklee@rippleai.co

Youngjae Yu
RippleAI & SNU
Seoul, Korea
yj.yu@rippleai.co

Gunhee Kim
RippleAI & SNU
Seoul, Korea
gunhee@snu.ac.kr

Abstract

We present a novel data augmentation technique, CRA (Contextual Response Augmentation), which utilizes conversational context to generate meaningful samples for training. We also mitigate the issues regarding unbalanced context lengths by changing the input-output format of the model such that it can deal with varying context lengths effectively. Specifically, our proposed model, trained with the proposed data augmentation technique, participated in the sarcasm detection task of FigLang2020, have won and achieves the best performance in both Reddit and Twitter datasets.

1 Introduction

The performance of many NLP systems largely depends on their ability to understand figurative languages such as irony, sarcasm, and metaphor (Pozzi et al., 2016). The results from the Sentiment Analysis task held in SemEval-2014 (Martínez-Cámara et al., 2014), for example, show that apparent performance drops occur when the figurative language is involved in the task. This work aims, in particular, to design a model that identifies sarcasm in the conversational context. More specifically, the goal is to determine whether a response is sarcastic or not, given the immediate context (*i.e.* only the previous dialogue turn) and/or the full dialogue thread (if available). For evaluation of our model, we participated in the FigLang2020 sarcasm challenge[1], and have won the competition as our model is ranked 1 out of 35 teams for the Twitter dataset and 1 out of 34 teams for the Reddit dataset.

We summarize our technical contributions to win the challenge as follows:

1. We propose a new data augmentation technique that can successfully leverage the structural patterns of the conversational dataset. Our technique, called CRA(Contextual Response Augmentation), utilizes the conversational context of the unlabeled dataset to generate new training samples.

2. The context lengths (*i.e.* previous dialogue turns) are highly variable across the dataset. To cope with such imbalance, we propose a context ensemble method that exploits multiple context lengths to train the model. The proposed format is easily applicable to any Transformer (Vaswani et al., 2017) encoders without changing any model architecture.

3. We propose an architecture where the Transformer Encoder is stacked with BiLSTM (Schuster and Paliwal, 1997) and NeXtVLAD (Lin et al., 2018). We observe that NeXtVLAD, a differentiable pooling layer, proves more effective than simple non-parametric mean/max pooling methods.

2 Approach

The task of our interest is, given response (r_1) and its previous conversational context ($c_1, c_2, \cdots, c_n$), to predict whether the response r_1 is sarcastic or not (See an example in Figure 2). We below discuss our model (section 2.1), training details (section 2.2) and the proposed data augmentation techniques (section 2.3).

	$train_{data}$	$valid_{data}$	$test_{data}$
Twitter	4000	1000	1800
Reddit	3520	880	1800

Table 1: Dataset Splitting.

---

[1] https://competitions.codalab.org/competitions/22247.
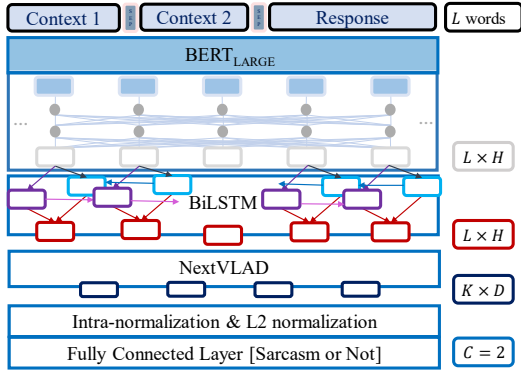
Figure 1: The architecture of our best performing model for sarcasm detection.

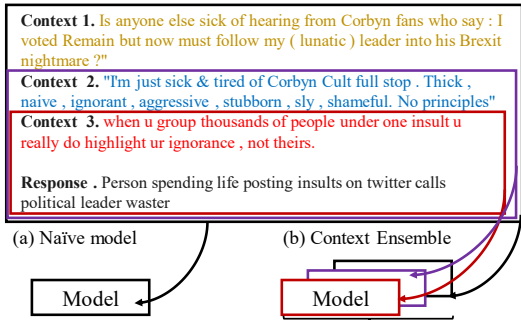
(a) Naïve model   (b) Context Ensemble

Figure 2: Illustration of the context ensemble method for Sarcasm detection. We train multiple models with different context window sizes, and ensemble them for inference.

2.1 The Model

Figure 1 describes the architecture of our best performing model. The model broadly consists of two parts: the transformer (BERT) (Devlin et al., 2018) and pooling layers, which are decomposed into BiLSTM (Schuster and Paliwal, 1997) and NetXtVLAD (Lin et al., 2018) as an improved version of NetVLAD (Arandjelovic et al., 2016). Reportedly, NetVLAD is a CNN-model that is highly effective and more resistant to over-fitting than usual temporal models such as LSTM or GRU (Lin et al., 2018). The Implementation of these models are as follows:

- BERT(large-cased): 24-layer, 1024-hidden and 16-heads.

- BiLSTM: 2-layer, 1024-hidden and 0.25-dropout.

- NeXtVLAD: 8-groups, 4-expansion, 128-number of clusters and 512-cluster size.
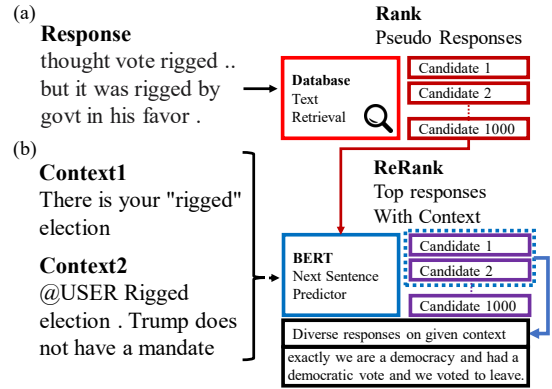

Figure 3: Overview of the proposed Contextual Response Augmentation (CRA). Using (a) Text query retrieval on sarcasm database and (b) Reranking best responses conditioned on a given context, we obtain various pseudo responses that are useful for training.

2.2 Training Details

We use the entropy loss on the last softmax layer in the model. The training batch size is 4 for all the experiments. We adopt the cyclic learning rate (Smith, 2017), where the initial learning rate is 1e-6, and the moment parameters are (0.825, 0.725).

Dataset Splitting. We further split the provided training set ($training_{data}$) into the training ($train_{data}$) and validation ($valid_{data}$) set as in Table 1. We use $valid_{data}$ for early stopping and the model performance validation during the training phase.

Context Ensemble. Figure 2 depicts the idea of the context ensemble method to cope with highly variable context lengths in the dataset. Instead of using the training data as their original forms only (Figure 2(a)), we consider multiple context window sizes as separate data, which can naturally balance out the proportion of short and long context (Figure 2(b)).

2.3 Data Augmentation

Van Hee et al. (2018) and Ilić et al. (2018) have observed that in the case of Twitter, fueling additional data from the same domain did not help much the performance for detecting sarcasm and irony. However, this does not mean that the data augmentation would fail to improve sarcasm detection. We use two techniques to augment the training data according to whether the data are labeled or not. Especially, our data augmentation method named Contextual Response Augmentation (CRA) can take advantage of unlabeled dialogue threads, which are abundant and cheaply collectible. Fig-

13

An unsarcastic sample	
c1	Dont mind me, its just a gun
c2	The dude in the front row is like 'Are we gonna do something?' It's the perfect example of how the bystander effect works, even amongst police (or whatever they are)
c3	100% if they were alone and saw this they'd doay something to the guy. But together you get a pack mentality.
r1	Unfortunately, the police are sometimes the victimizers
A sarcastic sample	
c1	Trump won Wisconsin by 27,000 votes. 300,000 voters were turned away by the states strict Voter ID law. There is your ÿriggedëlection ."
c2	@USER Rigged election . Trump does not have a mandate . Period.
r1	@USER @USER @USER exactly we are a democracy and had a democratic vote and we voted to leave

Table 2: Samples generated from unlabeled dataset

Metric	Precision		Recall		F1	
dataset	$twitter_{valid}$	$reddit_{valid}$	$twitter_{valid}$	$reddit_{valid}$	$twitter_{valid}$	$reddit_{valid}$
T+BiLSTM+NeXtVLAD	0.8295	0.6414	0.8816	0.7867	0.8548	0.7067
T+BiLSTM+MaxPool	0.8558	0.6620	0.8182	0.7092	0.8366	0.6848
T+BiLSTM+MeanPool	0.7339	0.6881	0.8683	0.5837	0.7954	0.6316
T+NeXtVLAD	0.8163	0.5891	0.8785	0.7976	0.8462	0.6777
T+BiLSTM+NeXtVLAD	0.8747	0.6938	0.9219	0.8187	0.8977	0.7513
T+BiLSTM+MaxPool	0.8318	0.6624	0.8751	0.7910	0.8529	0.7210
T+BiLSTM+MeanPool	0.7856	0.6089	0.8792	0.8070	0.8298	0.6941
T+NeXtVLAD	0.8525	0.6888	0.9101	0.7792	0.8804	0.7313

Table 3: Sarcasm detection performance on the validation set. The upper and lower part of the table respectively denote the performance before and after data augmentation is applied. We set the context length to 3 for all models.

ure 3 illustrates the overview of our CRA method whose details are presented in section 2.3.2.

2.3.1 Augmentation with Labeled Data

Each training sample consists of contextual utterances, a response and its label ("SARCASM" or "NOT_SARCASM"): $[c_1, c_2, \cdots, c_n, r_1, l_1]$. Our idea is to take the context sequence $[c_1, c_2, \cdots, c_n]$ as a new datapoint and label it as "NOT_SARCASM". As shown in Figure 2, without the response $[r_1]$, the sequence could not be labeled as "SARCASM". We hypothesize that these newly generated negative samples help the model better focus on the relationship between the response $[r_1]$ and its contexts $[c_1, c_2, \cdots, c_n]$. Also, we balance out the number of negative samples by creating positive samples via back-translation methods (Bérard et al. (2019); Zheng et al. (2019)), which simply translate the sentences into another language and then back to the original language to obtain possibly rephrased data points. For the back-translation, we have used 3 languages [French, Spanish, Dutch].

2.3.2 Augmentation with Unlabeled Data

We also generate additional training samples using the unlabeled data: $[c_1, c_2, \cdots, c_n, r_1]$. This approach is tremendously useful since a huge amount of unlabeled dialogue threads can be collected at little cost. As shown in Figure 3, the procedures for unlabeled augmentation are as follows:

1. We encode each response in the labeled training set using the BERT trained on natural inference tasks (Reimers and Gurevych, 2019).

2. Given unlabeled data $[c_1, c_2, \cdots, c_n, r_1]$, we encode $[r_1]$ and find the most similar top $k(=1000)$ data from the labeled database. We denote them as $\{r_{t,1}, \cdots, r_{t,k}\}$.

3. We rank the top k candidates according to the next sentence prediction (NSP) confidence of BERT[2]. That is, we input $[c_1, c_2, \cdots, c_n, sep, r_{t,i}]$ to BERT, and compute the NSP confidence of $r_{t,i}$ for all $i \in \{1, \cdots, k\}$. We then select the most confident response r_t^* with its label l_t^* and make a new data point $[c_1, c_2, \cdots, c_n, r_t^*, l_t^*]$.

Table 2 shows some samples generated from this technique. The quality of generated data depends undoubtedly on the degree of contextual conformity and similarity between the initial responses. We find, however, that adding more data makes the quality of the augmented data better as the label transfer noise becomes attenuated. In summary, besides the standard datasets shown in Table 7, we

---

[2] We fine-tune BERT only for the next sentence prediction task using the corpora in Table 7 and the $training_{data}$

Teams	Precision	Recall	F1
miroblog	**0.932**	**0.936**	**0.931**
nclabj	0.792	0.793	0.791
Andy3223	0.7910	0.7940	0.790
DeepBlueAI	0.78	0.785	0.779
ad6398	0.773	0.775	0.772
miroblog	**0.834**	**0.838**	**0.834**
Andy3223	0.751	0.755	0.75
DeepBlueAI	0.749	0.750	0.749
kevintest	0.746	0.746	0.746
Taha	0.738	0.739	0.737

Table 4: The FigLang2020 Sarcasm Scoreboard for Twitter (upper) and Reddit (below) dataset. Our method miroblog achieves the best performance in both datasets with significant margins.

$twitter_{valid}$	Precision	Recall	F1
no augmentation	0.8294	0.8816	0.8547
labeled augmentation	0.8676	0.8550	0.8613
unlabeled augmentation	0.8747	0.9219	0.8977

Table 5: Sarcasm detection performance according to data augmentation on the $twitter_{valid}$ dataset.

further crawled 100,000 texts from both Twitter and Reddit for the augmentation with unlabeled data.

3 Experiments

We first report the quantitative results by referring to the statistics in the official evaluation server [3] of the FigLang2020 sarcasm challenge as of the challenge deadline (*i.e.* April 16, 2020, 11:59 p.m. UTC). Table 4 summarizes the results of the competition, where our method named miroblog shows significantly better performance than other participants in both Twitter and Reddit dataset. We report Precision (P), Recall (R), and F1 scores as the official metrics.

3.1 Further Analysis

We perform further empirical analysis to demonstrate the effectiveness of the proposed ideas. We compare different configurations of pooling layers, context ensemble, and data augmentation.

Pooling Layers. Table 3 shows the comparison of sarcasm detection performance between NeXtVLAD and other pooling methods in performance. When coupled with BiLSTM, NeXtVLAD achieves better performance than max, and mean pooling methods.

Context Ensemble. Table 6 shows the comparison with different context ensemble methods.

---

[3]https://competitions.codalab.org/competitions/22247.

$twitter_{valid}$	Precision	Recall	F1
Ensemble (max context)	0.8558	0.8182	0.8366
Ensemble (3 context)	0.8320	0.8288	0.8304
Single (3 context)	0.8147	0.8052	0.8099

Table 6: Sarcasm detection performance according to the ensemble methods on the $twitter_{valid}$ dataset.

Reference	Name	Size
Ptáček et al. (2014)	Platek	57041
Riloff et al. (2013)	Riloff	1570
Khodak et al. (2017)	SARC-v2	321748
Khodak et al. (2017)	SARC-v2-pol	14340
Van Hee et al. (2018)	SemEval-2018-irony	3851
-	Web Crawled	100000

Table 7: The standard datasets and the crawled dataset (for unlabeled augmentation) used in the experiments.

We use the baseline (Transformer+BiLSTM+ Maxpooling) and train it without augmenting the training set. F1 scores of the model are better in the order of (a) ensemble with maximum context, (b) ensemble with three contexts and (c) no context. The performance gap with or without context ensemble implies that balancing out the samples in terms of context length is important. On the other hand, the performance gap between (a) and (b) is only 0.006, indicating that the use of older than three recent conversational contexts is scarcely helpful.

Data Augmentation. Table 5 compares the sarcasm detection results when the data augmentation is applied or not. The augmentation with labeled data increases the F1 score from 0.854 to 0.861. The augmentation with unlabeled data further enhances performance from 0.861 to 0.897. The results demonstrate that both augmentation techniques help with the performance.

3.2 Error Analysis

In order to better understand when our data augmentation methods are effective, we further analyze some examples of the following three cases according to whether the proposed labeled and unlabeled data augmentation (DA) is applied or not: (i) the prediction is wrong without DA but correct with DA, (ii) the prediction is correct without DA but wrong with DA, and (iii) the prediction is wrong with and without DA. In other words, (i) is the case where DA helps, (ii) is the one where DA hurts, and (iii) is the one where DA fails to improve.

Table 8 shows some examples of these three cases. (i) The initial steps of the CRA involve finding similar training samples from the labeled database. Thus, after applying CRA, samples containing specific hashtags, *e.g.* #NotReally #Relax,

(i) The prediction is wrong without DA but correct with DA.	
c1	Any practice could be anyone's last practice. Yes.
c2	@USER report: tom brady struck by lighting after leaving practice.
r1	**[SARCASM]** @USER Report: Tom Brady abducted by space aliens during practice. #NotReally #Relax
(ii) The prediction is correct without DA but wrong with DA	
c1	@USER @USER @USER The racist trump is a Russian puppet. He's a loser who's trying to destroy our constitution and hand this Country over to Putin. He steals with the help of his white nationalist supporters. He should be removed from Office and put in prison.
c2	@USER @USER @USER And who's drinking the koolaide ? Mueller said no collusion or obstruction after spending $ 30 million investigating - with full access to the White House. White Nationists unsubstantiated conspiracy theory. Trump will win 2020 because people see him succeed through the nonsense.
r1	**[NOT_SARCASM]** @USER @USER @USER You didn't bother to read the Mueller report, did you? It was Barr who falsely exonerated your beloved cult leader. Read the Mueller report. Until you do, don't propagate this lie. Educate yourself and read the report or shut up. You ' ll believe anything except the truth.
r2	**[SARCASM]** you not worry i are so blind, deaf. I KNOW you have lost your sight (with regard) listened to your cult leaders and Faux News and some Republicans.
(iii) The prediction is wrong with and without DA.	
c1	I love this land called America #VPDExperiment #VPDDay @USER and @USER at @USER. The 30 Best Things to do in Washington DC: URL
c2	@USER @USER @USER Makes me just want to bow out of this whole thing right now... LOL @USER @USER @USER Noooooo! It's just the way I edit.
c3	I'm trying all sorts of styles this 30 days. No competition being done.
r1	**[SARCASM]** @USER @USER @USER Sorry, I forgot to use the font! I'm loving your videos. Its giving me ideas and inspiration for some stuff I'd like to try.

Table 8: Examples of three cases where data augmentation helps, hurts, or fails to improve the sarcasm predction.

are included in the training set. We observe that theses tags tend to occur with the samples that are labeled "SARCASM", and thus CRA helps the model learn the correlation between the hashtags and the labels. (ii) The augmented response ($r2$) contains the phrase "cult leader" as in the original response ($r1$). The corresponding label, however, is "SARCASM". When the newly added samples do not match the context, or the labels are incorrect, CRA degrades the prediction. (iii) The third case arises mostly when the situation is subtle and requires external knowledge beyond the given context. In order for the model to correctly classify the response as "SARCASM", the model requires to understand the tag #VPD(Video Per Day). It is not clear what #VPD is from the context, and without such knowledge, the model may still make incorrect predictions.

4 Conclusion

We proposed a new data augmentation technique, CRA (Contextual Response Augmentation), that utilizes the conversational context of the unlabeled data to generate meaningful training samples. We demonstrated that the method boosts the perfor-

mance of sarcasm detection significantly. The employment of both augmentations with labeled and unlabeled data enables the system to achieve the best F1 scores to win the FigLang2020 sarcasm challenge on both datasets of Twitter and Reddit.

References

Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. 2016. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307.

Alexandre Bérard, Ioan Calapodescu, and Claude Roux. 2019. Naver labs europe's systems for the wmt19 machine translation robustness task. In *arXiv preprint arXiv:1907.06488*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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

Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli.

2017. A large self-annotated corpus for sarcasm. *arXiv preprint arXiv:1704.05579*.

Rongcheng Lin, Jing Xiao, and Jianping Fan. 2018. Nextvlad: An efficient neural network to aggregate frame-level features for large-scale video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0.

Eugenio Martínez-Cámara, M Teresa Martín-Valdivia, L Alfonso Urena-López, and A Rturo Montejo-Ráez. 2014. Sentiment analysis in twitter. *Natural Language Engineering*, 20(1):1–28.

Federico Alberto Pozzi, Elisabetta Fersini, Enza Messina, and Bing Liu. 2016. *Sentiment analysis in social networks*. Morgan Kaufmann.

Tomáš Ptáček, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on czech and english twitter. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 213–223.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–714.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

Leslie N Smith. 2017. Cyclical learning rates for training neural networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 464–472. IEEE.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Renjie Zheng, Hairong Liu, Mingbo Ma, Baigong Zheng, and Liang Huang. 2019. Robust machine translation with domain sensitive pseudo-sources: Baidu-osu wmt19 mt robustness shared task system report. *arXiv preprint arXiv:1906.08393*.