Sarcasm Detection: Using BERT along with Data Augmentation and Sarcasm Generator

Kaustubh. M. Prabhu

CSE 5095 Fall’21

University of Connecticut.

kaustubh.prabhu@uconn.edu

***Abstract-* We present a novel approach for detecting sarcasm by combining two completely different approaches which is using BERT along with Contextual Response, and more features generated using Sarcasm Generator whose results are considered for final comparison context. This method in the generator employs a retrieve and edit framework to consider two of the most important characteristics of sarcasm: reversal of valence along with the semantic incongruity of context, which includes a commonsense or knowledge bank. This combined approach mitigates the issues regarding unbalanced context lengths and the linguistic features of the statement.**

**1. INTRODUCTION**

The performance of a lot of the existing NLP systems largely depends on their ability to understand the figurative aspects of language such as sarcasm, irony, metaphor (Pozzi et al., 2016). In SemEval-2014 the paper (Martınez-Ca ́mara et al., 2014), shows that apparent performance drop occurs when the aspects of figurative language are involved at task at hand. This work aims to design a model that identifies sarcasm not only in the conversational context but also linguistic sense. The goal is to determine if a response is sarcastic or not, given the immediate context of previous two statements.

The technical contributions are summarized as follows:

1. We use the conversational context of the dataset and enter it into a sarcasm generation module that responds sarcastically to input. The module implements eight sarcasm generators, each covering a peculiar form of sarcasm. (Aditya Joshi et al., 2015)
2. The similarity scores of these eight sarcastic responses along with the results of classification from BERT are used as the final measures to determine the class.

|  |  |  |  |
| --- | --- | --- | --- |
|  | traindata | validdata | testdata |
| Twitter | 4000 | 1000 | 1800 |
| Reddit | 3250 | 880 | 1800 |

Table 1: Dataset Splitting.

Sarcasm has been widely studied in linguistics. Different kinds of sarcasm and different dimensions associated with sarcasm have been defined. (E. Camp, 2012) (J. D. Campbell

et al., 2012). The motivation here is sarcasm generation will lead to a deeper understanding of the phenomenon of sarcasm, and as a result, help sarcasm detection as well. (Aditya Joshi et al., 2015)

* 1. **Dataset**

**1.1.1 Reddit Training Dataset**

SARC : Khodak et al. (2017) introduced the self-annotated Reddit Corpus which is a very large collection of sarcastic and non-sarcastic posts curated from different subreddits of varied topics. It has self-labeled sarcastic posts where users have labeled their posts as sarcastic by marking “/s” to the end of sarcastic posts which is known sign for marking sarcasm on reddit. For any such sarcastic post, the corpus also provides the full conversation context, i.e., all the prior turns that took place in the dialogue. This dataset chose sarcastic responses with at least two prior turns. Note, for many responses in the training corpus the number of turns is much more. Same was for testing and validation dataset. This is from the shared task from workshops in FigLang 2020.

**1.1.2 Twitter Training Dataset**

For the Twitter dataset, hashtags were used for annotations. The sarcastic tweets were collected using hashtags: *#sarcasm* and *#sarcastic*. Sentiment tweets were considered non sarcastic, i.e., the methodology proposed in related work (Muresan et al., 2016) used. We rely on *self-labeled* tweets, thus, it is always possible that sarcastic tweets were mislabeled with sentiment hashtags or users did not use the #sarcasm hashtag at all. Hence after manual evaluation of around 200 sentiment tweets very few such cases. So the data was selected. This was also from FigLang 2020.

https://competitions.codalab.org/ competitions/22247

*Proceedings of the Second Workshop on Figurative Language 2020*

Association for Computational Linguistics https://doi.org/10.18653/v1/P17

12

**2. APPROACH**

The task of our interest is, given response (r1) and its previous conversational context (c1 , c2), to predict whether the response r1 is sarcastic or not. The modules below discuss our model (section 2.1), training details (section 2.2), the data augmentation technique (section 2.3), the sarcasm generator module (section 2.4) and the ensemble model (section 2.5)

**2.1 The Model**

The model broadly consists of two parts: the transformer (BERT) (Devlin et al., 2018) and pooling layers, which are just max pooling. BERT(large-cased): 24-layer, 1024-hidden and 16-heads.

**2.2 Training Details**

For the last softmax layer in the model entropy loss was used. The training batch size is 4. The cyclic learning rate (Smith, 2017), where the initial learning rate is 1e- 6, and the moment parameters are (0.825, 0.725) was adopted. (Hankyol Lee et al. 2020).

*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.

**2.3 Data Augmentation**

Hankyol Lee et al. (2020) showed that their data augmentation method which they named as Contextual Response Augmentation (CRA) can take advantage of unlabeled dialogue threads, which are abundant and cheaply collectible.

**2.3.1 Augmentation with Labeled Data**

Each of the sample in the training data consists of contextual utterances, it’s response and its label whether the response is sarcastic or not. (”SARCASM” or ”NOT SARCASM”): [c1, c2, · · · , cn, r1, l1].

The idea used here is to take the context sequences [c1,c2,···,cn] as a totally new datapoint and label it as ”NOT SARCASM”. As shown in Figure 2 without the presence of response [r1], the sequence could not be labeled as “SARCASM”. This is being considered as a negative sample to have a better connection in between response and the given or considered. context.

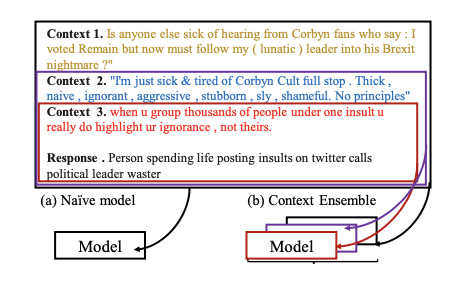


Figure 2: Illustration of the context ensemble method for Sarcasm detection. ***Hankyol Lee et al. (2020).***

Also, the task of balancing out the number of negative samples by adding newly formulated positive samples using the back-translation methods (Berard 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.

**2.3.2 Augmentation with Unlabeled Data**

Additional training samples were also generated using the unlabeled data: [c1,c2,···,cn,r1]. This approach is found to be extremely useful as it generates huge amount of unlabeled dialogue threads can be collected at little cost.

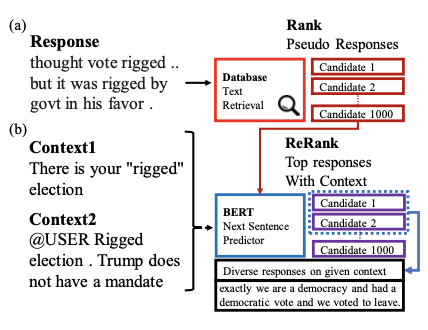


Figure 3: Overview of Contextual Response Augmentation (CRA). Using (a) Text query retrieval on sarcasm database and (b) Reranking the best responses conditioned on context. ***Hankyol Lee et al. (2020).***

As shown in Figure 3, the procedures for unlabeled augmentation are as follows:

1. Each response in the labeled training set is encoded using the BERT trained on natural inference tasks (Reimers and Gurevych, 2019).

2. Given that the unlabeled data [c1, c2, · · ·, cn, r1], [r1] is encoded then used to find the most similar top k where (k = 1000) data from the labeled database.

3. The top ranked candidates are generated according to the next sentence prediction (NSP) confidence of BERT. Then the most confident response rt∗ with its label to create a new datapoint.

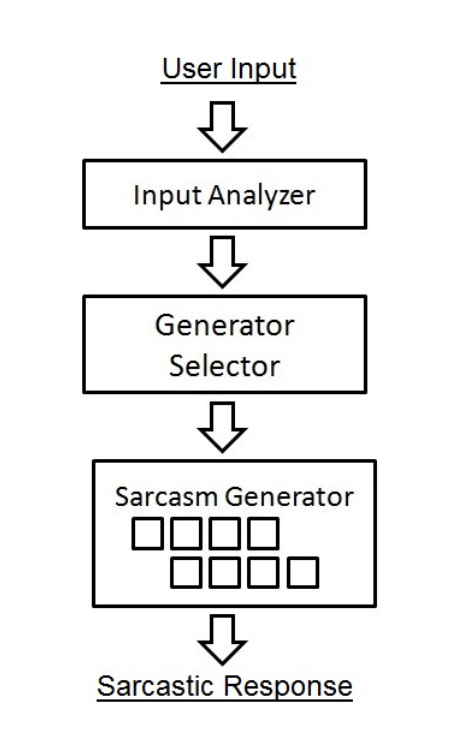


Figure 4: Overview of SarcasmBot Architecture ***Aditya Joshi et al., 2015***

**2.4 Sarcasm generator**

This module is direct implementation of ‘SarcasmBot’ (Aditya Joshi et al., 2015). The user input and sarcastic response are syntactically valid English sentences, where the user input is first analyzed using the Input Analyzer, then sent to the Sarcasm Generator type selector. The generator type selector chooses among eight Sarcasm Generators. Explained in table 2. The chosen sarcasm generator is used to then produce the sarcastic response. All three modules are rule-based, relying on lexicons. The input for the generator model is the latest context.

|  |  |
| --- | --- |
| Sarcasm Generator | Description |
| Offensive Word Response | Generator In case an offensive word is used in user input, select a placeholder from a set of responses. |
| Opposite Polarity Verb-Situation Generator | Randomly select a verb. Compute its sentiment. Discover a situation which is opposite in sentiment. |
| Opposite Polarity Person-Attribute Generator | Randomly select a named entity. Select incongruent pairs of famous people |
| Irrealis Sarcasm Generator | Create a hypothetical situation that is impossible by selecting from a set of undesirable situations. |
| Hyperbole Generator | Select a noun phrase in the user input. Generate a hyperbole with a ‘best ever’ style regular expression. |
| Incongruent Reason Generator | Select an unrelated reason as a response for a user input. |
| Sentiment-based Sarcasm Generator | Compute sentiment of user input. Generate a response opposite in sentiment |
| Random Response Generator | Select one positive exclamation and one negative exclamation randomly from a set of exclamations. Place them together. |

Table 2: A Tabulated Summary of Sarcasm Generators in SarcasmBot

**2.5 Ensemble Model**

We use a simple business logic algorithm to qualify our classification results as sarcastic or not. We take the output of BERT with it’s confidence score and then ***cosine similarity*** of all the generated 8 sarcasm responses and for all the results of BERT where confidence is less than 0.8 we see if the similarity scores of any of the generated responses to the actual given response are more than 0.75 then we qualify that case as a Sarcastic comment.

We also evaluate just the generated responses for detection of sarcasm by assessing whether any of them had a similarity score of more than 0.75 for the response under consideration.

**3. EXPERIMENTS**

We ran four different methods for a comparative analysis of our results and gain insights. The first case was a BERT model with the variable context input for training, the second was the aforementioned BERT along with data augmentation which was discussed previously. The third method was the evaluation of generators and the final was the ensemble model. The context size of all the BERT was set to 3. The results are seen in table 4.

The evaluation of all the models was done using standard metrics of F1 score, Precision and Recall. Weighted averages were considered.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Metric | Precision | | Recall | | F1 | |
| Dataset | Tweet | Reddit | Tweet | Reddit | Tweet | Reddit |
| BERT | 0.735 | 0.75 | 0.77 | 0.72 | 0.74 | 0.725 |
| BERT + Augment | 0.81 | 0.79 | 0.813 | 0.804 | 0.825 | 0.813 |
| Sarcasm similarity | 0.671 | 0.70 | 0.691 | 0.68 | 0.667 | 0.683 |
| Ensemble Model | 0.802 | 0.81 | 0.82 | 0.814 | 0.821 | 0.826 |

Table 4: Sarcasm detection performance on test set.

**4. CONCLUSION & FUTURE WORK**

We proposed a new combination model which uses data augmentation technique, CRA (Contextual Response Augmentation), that utilizes the conversational context of the unlabeled data to generate meaningful training samples and Sarcasm Generators. This definitely opened up potential avenues in research for considering the sarcasm generation as a potential linguistic feature analysis to be used as an input towards the analysis for sarcasm detection.

Potential future work in this area will be using newer and better Sarcasm Generators with the current use logic. Along with that the output from sarcasm generation can be used as a positive sample generation for training the models.

**5. REFERENCES**

Lee, H., Yu, Y., Kim, G. 2020: Augmenting data for sarcasm detection with unlabeled conversation context. In: FigLang, pp. 12–17

Joshi, A. 2015. SarcasmBot : An open-source sarcasm-generation module for chatbots.

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 Be ́rard, 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 Ilic 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*.

F. Bessho, T. Harada, and Y. Kuniyoshi. Dialog system using real-time crowdsourcing and twitter large-scale corpus. In Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGDIAL ’12, pages 227–231, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

D. G. Bobrow, R. M. Kaplan, M. Kay, D. A. Norman, H. Thompson, and T. Winograd. Gus, a frame-driven dialog system. Artificial intelligence, 8(2):155–173, 1977.

E. Camp. Sarcasm, pretense, and the semantics/pragmatics distinction\*. Noˆus, 46(4):587–634, 2012

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

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 Mart ́ınez-Ca ́mara, M Teresa Mart ́ın-Valdivia, L Alfonso Urena-Lo ́pez, and A Rturo Montejo-Ra ́ez. 2014. Sentiment analysis in twitter. *Natural Lan- guage Engineering*, 20(1):1–28.

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

Toma ́sˇ Pta ́cˇek, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on czech and english twitter. In *Proceedings of COLING 2014, the 25th Inter- national 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 Lan- guage Processing*, pages 704–714.