TechSSN at SemEval-2022 Task 6: Intended Sarcasm Detection using Transformer Models

Rajalakshmi Sivanaiah, Angel Deborah S, Sakaya Milton R,
Mirnalinee T T, Ramdhanush Venkatakrishnan
Department of Computer Science and Engineering
Sri Sivasubramaniya Nadar College of Engineering
Chennai - 603110, Tamil Nadu, India
{rajalakshmis, angeldeborahs}@ssn.edu.in,
{miltonrs, mirnalineett,ramdhanush2010105}@ssn.edu.in

Abstract

Irony detection in the social media is an upcoming research which places a main role in sentiment analysis and offensive languague identification. Sarcasm is one form of irony that is used to provide intended comments against realism. This paper describes a method to detect intended sarcasm in text (SemEval-2022 Task 6). The TECHSSN team used Bidirectional Encoder Representations from Transformers (BERT) models and its variants to classify the text as sarcastic or non-sarcastic in English and Arabic languages. The data is preprocessed and fed to the model for training. The transformer models learn the weights during the training phase from the given dataset and predicts the output class labels for the unseen test data.

1 Introduction

Sarcasm is a form of verbal irony that occurs when there is a discrepancy between the literal and intended meanings of an utterance. This is often used to express the opposite meaning of the words spoken. This is used frequently while making fun of someone or something, and is used in a variety of contexts, like casual conversation, memes, or even public speaking, to convey a variety of meanings, providing a certain level of depth and sophistication to the communication of the language.

Sarcasm is present in all overcontemporary social media networks and may reduce the efficiency of systems that perform operations on these sarcastic data such as sentiment analysis, opinion mining, author profiling, and harassment detection (Liu, 2012; Rosenthal et al., 2014; Maynard and Greenwood, 2014; Van Hee et al., 2018). It generates misleading conclusions, due to its nature to imply different meaning than what is intended on the surface. Even in SemEval, (Rosenthal et al., 2014) shows that there is a significant drop in system performance when processing sarcastic text data, in comparison to non-sarcastic data. These systems are used in industry, driving marketing, administration, and investment decisions (Medhat et al., 2014). This clearly shows that developing models to find and detect sarcasm is becoming more important by the day.

The iSarcasmEval Task 6 for SemEval 2022 (Abu Farha et al., 2022) is comprised of three SubTasks: To classify the input text as sarcastic or not, in English (SubTaskA English) and Arabic (SubTask A Arabic), and further classify sarcastic text into categories (SubTask B), and given two phrases with same meaning, identify the sarcastic one (SubTask C English, SubTask C Arabic). Of these, the TechSSN team has attempted to solve SubTaskA English, SubTask A Arabic, and SubTask B.

2 Related Work

A lot of the previous sarcasm detection datasets have been annotated using a weak supervision method. In weak supervision, text data is classified as sarcastic only if it meets a certain set of conditions that are decided upon prior to the collection and analysis of the data. This includes using tags (e.g. #sarcasm, #irony) (Ptacek et al., 2014; Khodak et al., 2018) to perform the above mentioned classification. However, this can result in noisy labels for many reasons, as demonstrated by (Oprea and Magdy, 2020).

Other work makes use of manual labelling, where sarcasm labels are provided by human annotators (Filatova, 2012; Riloff et al., 2013a; Abercrombie and Hovy, 2016). But, this can mean that labels are subjective in nature, i.e. labels may reflect annotator perception, which may differ from the meaning intended by the author, as pointed out by (Oprea and Magdy, 2020).

Moreover, a lot of sarcasm detection work applies only to the English language and, because of the socio-cultural aspects of sarcastic communication (Oprea and Magdy, 2020), it is doubtful that

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 851 - 855

the models trained to detect sarcasm in the English language could do the same task with the same effectiveness on other languages such as Arabic (where most of the sarcasm detection is carried out using the above-mentioned weak supervision).

We have participated for irony and sarcasm detection SemEval task in (Sivanaiah et al., 2018) and used MultiLayer Perceptron model to find the ironic and sarcastic tweets. We have used CNN, RNN, LSTM, BERT and COLBERT models for offensive language detection in earlier SemEval workshop tasks (Sivanaiah et al., 2021), (Sivanaiah et al., 2020), (Sivanaiah et al., 2019) in which BERT models provides better results than other machine learning and deep learning models.

3 Methodology

3.1 Dataset

We used the dataset provided by the organizers of the Task to train and build the model. The dataset has fields for sarcasm, irony, satire, understatement, overstatement, rhetorical questioning, all of which are binary, and other categories of sarcastic text, along with a field for a non-sarcastic rephrase. It contains about 3467 entries of which about 866 are sarcastic and the rest are not sarcastic. The Arabic dataset has fields for sarcasm, rephrased text, and the regional dialect. It has about 3102 entries of which about 746 entries are sarcastic and the rest are not sarcastic. The test dataset entries for Task A English is 1400, Task A Arabic is 1400, Task B is 1616. In addition, the Task B dataset has fields for sarcastic rating and regional dialect.

3.2 Data Pre-processing

First, the raw data is tokenized. This means that each sentence is tokenized or split into sub-words for the BERT model. This is done using thecompute_input_arrays method that is available under the BertTokenizer class. This method makes use of a pre-trained 'BERT-base-uncased' model to tokenize the input sentences. The maximum sequence length is set as 200 as BERT requires inputs to be in a fixed size and shape. After trimming the input, the pre-trained model combines segments and creates appropriate masks for the given data. The input representation for the model is shown in Figure 1. The token embeddings, segmentation embeddings and position embeddings are summarized together to form the input embeddings.

3.3 Models and training

We have used pre-trained models for each Subtask. We have used BERT and its modifications such as Contextualized Late Interaction over BERT (CoLBERT) (Khattab and Zaharia, 2020) and A Robustly Optimized BERT Pretraining Approach (RoBERTa) (Liu et al., 2019). BERT, simply put, is a stack of encoders part of the transformer architecture. It uses attention on the decoder side, and self-attention on the encoder side. Base BERT has 768 hidden units, 12 attention heads, and 110M parameters. Similarly, Large BERT has 1024 hidden units, 16 attention heads, and 340M parameters. The BERT model takes a classification token, followed by a sequence of words as input. The input is then passed through several layers of encoder stack (12 in Base BERT, 24 in Large BERT). Each of the many layers applies self-attention, sends the output through a feedforward network of hidden layers, and then sends the output to the next encoder layer.

The procedures for pretraining and finetuning the model is shown in Figure 2. Same architecture structure is used in pre-training anf fine-tuning, differs only in the output layers. Both the encoder and decoder stream tasks are initialized with the same pre-trained model parameters. All parameters are fine-tuned in tuning phase. [CLS] is a special symbol added in front of every input example, and [SEP] is a special separator token (e.g. separating questions/answers).

RoBERTa is a modification to the original BERT, and needs about the same amount of parameters that Base BERT requires (110M). It takes more training time than BERT, about 4-5 times more than BERT, but can provide more accurate results and predictions compared to BERT. CoLBERT is faster than many other BERT-based models and uses a pre-trained BERT model to handle late interactions. The model is trained with the training data provided by the organizers.

4 Results and Discussions

The test dataset was provided by the SemEval-2022 organizers and was given to different models for each Subtask and the results are listed in Tables 1 to 3.

ColBERT model gives better accuracy for the English language than Arabic. For subtask B we have used multilabel classifier to predict the output.

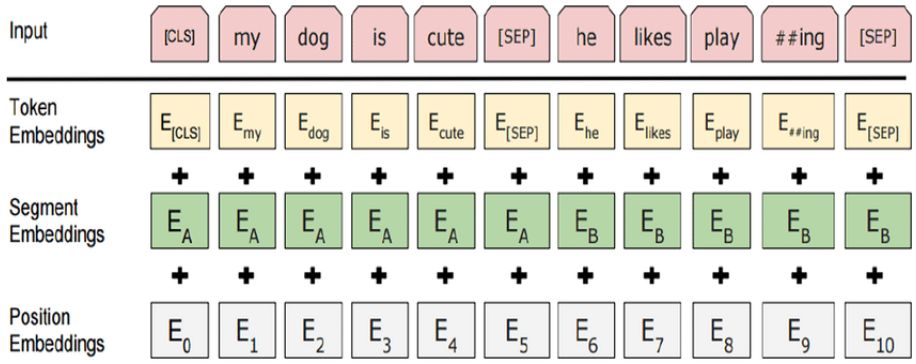

Figure 1: Input Representation – *source:(Devlin et al., 2018)*

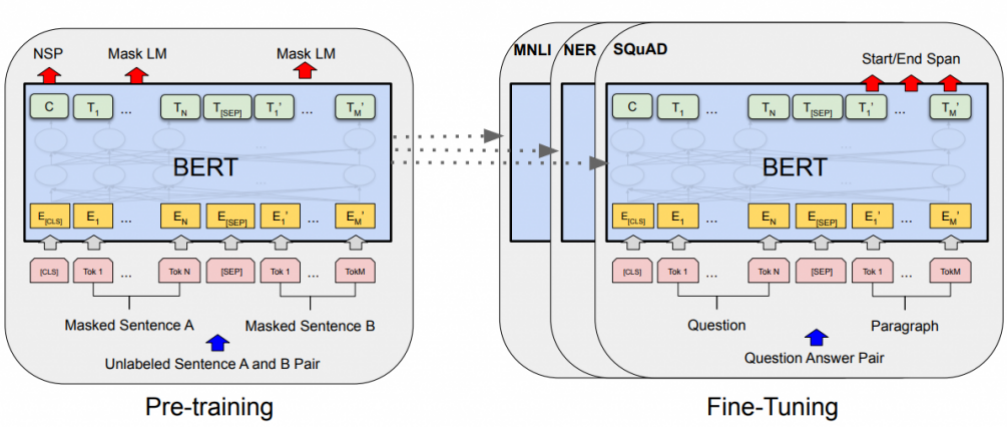

Figure 2: Pretraining and Fine tuning BERT – *source:(Devlin et al., 2018)*

Models	F1-Score	Accuracy
BERT	0.2558	0.2936
ColBERT	0.2637	0.7407

Table 1: Subtask A - English Results

Models	F1-Score	Accuracy
BERT	0.2292	0.3707

Table 2: Subtask A - Arabic Results

Models	F1-Macro
RoBERTa	0.0596

Table 3: Subtask B Results

5 Conclusion

It is obvious that the detection of sarcasm and its various categories is important as it is pivotal in computations like sentiment analysis, opinion mining, author profiling, or harassment checking. This can become increasingly difficult and tedious to compute, as sarcasm is extremely subjective as its nature itself is implying and contradictory, and the amount of data to be analyzed is getting larger and complex by the day. It is also a big step for Natural Language Processing (NLP) as it can tremendously help in the creation of more sophisticated virtual assistants and chatbots, which can emulate conversations that are closer to human interactions and life-like.

SemEval-2022 Task 6 is comprised of three SubTasks of which the TechSSN team has participated in Subtask A English, Subtask A Arabic and Subtask B. Deep learning models like BERT, RoBERT, and CoLBERT were used to carry out the tasks successfully. The team was able to obtain the 27th rank in Task A – English, 29th rank in Task A – Arabic, and 17th rank in Task B. Results show that BERT based models perform better on average than other conventional models like Logistic Regression Decision Tree, Support Vector Machines etc. Because of BERT's multilingual nature, sarcasm detection can be carried out in other languages across the globe. We would like to investigate further and apply these models to other languages. The accu-

racy could potentially be improved by using more advanced and efficient pre-processing techniques.

References

Gavin Abercrombie and Dirk Hovy. 2016. Putting sarcasm detection into context: The effects of class imbalance and manual labelling on supervised machine classification of twitter conversations. In *Proceedings of the ACL 2016 student research workshop*, pages 107–113.

Ibrahim Abu Farha, Silviu Oprea, Steven Wilson, and Walid Magdy. 2022. SemEval-2022 Task 6: iSarcasmEval, Intended Sarcasm Detection in English and Arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

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*.

Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Lrec*, pages 392–398. Citeseer.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Mikhail Khodak, Nikunj Saunshi, Yingyu Liang, Tengyu Ma, Brandon Stewart, and Sanjeev Arora. 2018. A la carte embedding: Cheap but effective induction of semantic feature vectors. *arXiv preprint arXiv:1805.05388*.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Diana G Maynard and Mark A Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Lrec 2014 proceedings*. ELRA.

Walaa Medhat, Ahmed Hassan, and Hoda Korashy. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal*, 5(4):1093–1113.

Preslav Nakov, Sara Rosenthal, Svetlana Kiritchenko, Saif M Mohammad, Zornitsa Kozareva, Alan Ritter, Veselin Stoyanov, and Xiaodan Zhu. 2016. Developing a successful semeval task in sentiment analysis of twitter and other social media texts. *Language Resources and Evaluation*, 50(1):35–65.

Silviu Oprea and Walid Magdy. 2019. Exploring author context for detecting intended vs perceived sarcasm. *arXiv preprint arXiv:1910.11932*.

Silviu Vlad Oprea and Walid Magdy. 2020. The effect of sociocultural variables on sarcasm communication online. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW1):1–22.

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.

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.

Rajalakshmi Sivanaiah, Angel Deborah S, S Milton Rajendram, and Mirnalinee T T. 2018. SSN MLRG1 at SemEval-2018 task 3: Irony detection in English tweets using MultiLayer perceptron. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 633–637, New Orleans, Louisiana. Association for Computational Linguistics.

Rajalakshmi Sivanaiah, Angel Deborah S, S Milton Rajendram, Mirnalinee TT, Abrit Pal Singh, Aviansh Gupta, and Ayush Nanda. 2021. TECHSSN at SemEval-2021 task 7: Humor and offense detection and classification using ColBERT embeddings. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 1185–1189, Online. Association for Computational Linguistics.

Rajalakshmi Sivanaiah, Angel Suseelan, Logesh B, Harshini S, Geetika B, Dyaneswaran S, S Milton Rajendram, and Mirnalinee T T. 2019. TECHSSN at SemEval-2019 task 6: Identifying and categorizing offensive language in tweets using deep neural networks. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 753–758, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Rajalakshmi Sivanaiah, Angel Suseelan, S Milton Rajendram, and Mirnalinee T.t. 2020. TECHSSN at SemEval-2020 task 12: Offensive language detection using BERT embeddings. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2190–2196, Barcelona (online). International Committee for Computational Linguistics.

Cynthia Van Hee, Gilles Jacobs, Chris Emmery, Bart Desmet, Els Lefever, Ben Verhoeven, Guy De Pauw,

Walter Daelemans, and Véronique Hoste. 2018. Automatic detection of cyberbullying in social media text. *PloS one*, 13(10):e0203794.