IISERB Brains at SemEval-2022 Task 6: A Deep-learning Framework to Identify Intended Sarcasm in English

Tanuj Singh Shekhawat[*1], **Manoj Kumar**[*2], **Udaybhan Rathore**[*3], **Aditya Joshi**[4], **Jasabanta Patro**[5]

[1235] Indian Institute of Science Education and Research, Bhopal

[4] SEEK, Australia

{[1]tanuj19, [2]manoj19, [3]udaybhan19, [5]jpatro}@iiserb.ac.in, [4]ajoshi@seek.com.au

Abstract

This paper describes the system and the resulted models submitted by our team "IISERB Brains" to SemEval-2022 Task 6 competition. We participated in the three sub-tasks for English language datasets. Our submitted models use BERT-based classifiers along with data augmentation. We ranked **19th** out of **43 teams** for sub-task A, **8th rank** out of **22 teams** for sub-task B, and **13th rank** out of 16 teams for sub-task C. In this paper, we describe details of our submissions, related evaluation and additional experiments conducted post the termination of the shared task. Our code and used additional resources are present in GitHub[1] for reproducibility. *Authors with equal contributions are marked by *.*

1 Introduction

Sarcasm in spoken or written form is a type of verbal irony that indicates the difference between the literal and intended meanings of an utterance (Joshi et al., 2017). While many people use it as a bitter remark to mock or ridicule a target(Patro et al., 2019), some also use it as a joke to amuse others(Joshi et al., 2015a). Sarcasm often used together or interchangeably with other ironic categories, is considered an essential component of human communication. A large portion of the web and social media text is sarcastic, which creates a challenge for traditional natural language processing (NLP) tasks like sentiment classification, opinion mining, harassment detection, author profiling etc. Since these systems are deployed widely across various industries, administration, data analysts etc, designing a robust sarcasm detecting component would help the downstream tasks substantially. The SemEval-2022 task 6(Abu Farha et al., 2022) identifies some of the challenges persisting till now, particularly in English and Arabic texts. Our team participated in the tasks floated for the English language under the name "IISERB Brains". The organizers have floated three sub-tasks: (i)Sub-task A to detect whether a given text is sarcastic, (ii)Sub-task B to identify which ironic category the sarcastic tweet belongs to, and (iii) Sub-task C to given two pieces of text, identify the sarcastic one.

In this paper, we report the details of our participating systems and their performance on the evaluation data. The paper also contains details of additional experiments that were a part of our participation in the shared task. The paper is organised as follows. Section 2 describes past work related to the sub-tasks, while section 3 presents the dataset and task details. Following that, Section 4 reports the details of our systems followed by experiments and results in Section 5. Finally, in section 6, we conclude the paper with a summary of the overall results.

With our submitted systems, we obtained the **19th rank** out of **43 teams** for sub-task A, **8th rank** out of **22 teams** for sub-task B, and **13th rank** out of 16 teams for sub-task C. Our code is made public to ensure the reproducibility of our results.

2 Related Work

Sarcasm has been an interesting research topic for computational linguists for a long time(Joshi et al., 2017; Pilling et al., 2017). Researchers have studied the topic from depth(sarcasm sub-categories, linguistic nuances etc.) and breadth (multi-modal sarcasm(Castro et al., 2019a), multi-lingual sarcasm (Liu et al., 2021; Bansal et al.), sarcasm targets (Patro et al., 2019), etc). A considerable amount of work has been reported on the task 'sarcasm detection' itself. This task particularly deals with identifying whether a given text is sarcastic. The sub-task A proposed by the organizers also belongs to this task.

[1] `https://github.com/manojmahan/iSarcasmEval-Intended-Sarcasm-Detection-In-English-main`

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 938 - 944
July 14-15, 2022 ©2022 Association for Computational Linguistics

While the traditional approaches have used feature-based machine learning models, recent approaches mostly relied on deep-neural models to report state of art performance. Lexical features like emoticons, special characters, word-patterns have always been preferred features along with semantic features like parts-of-speech tags, contrasting sentiments etc(Davidov et al., 2010)(Veale and Hao, 2010) (Riloff et al., 2013)(Joshi et al., 2015b)(Ghosh et al., 2018) in traditional approaches. Deep-neural methods, on the other hand, learn latent features for the same task. They rely on architectures like RNNs, CNNs, transformers etc. In their models (Joshi et al., 2017; Pilling et al., 2017; Tay et al., 2018; Tarunesh et al., 2021; Jaiswal, 2020). Pre-trained language models like BERT, RoBERTa, XLNet etc, have been extensively used as token-encoders in these models.

Identifying sarcastic intention has always been a challenging task, even humans sometimes have difficulties. Recently, researchers have started focusing on contextual information such as author context (Ghosh et al., 2020), multi-modal context (Ghosh et al., 2020), eye-tracking information (Govindan et al., 2018), or conversation context (Ghosh et al., 2020; Srivastava et al., 2020) to capture it. The sub-task A is related to shared tasks in the domain of figurative analysis such as a SemEval task on irony detection in Twitter that focuses on the utterances in isolation.

3 Background

3.1 Dataset details

We used the English dataset provided by the task organizers(Abu Farha et al., 2022). It has 3468 English tweets. It has 867 sarcastic and 2601 non-sarcastic tweets, which indicates that the dataset is highly unbalanced. For each sarcastic tweet, the organizers have also provided the ironic sub-classes to which the tweet belongs. The sub-classes are sarcasm, irony, satire, understatement, overstatement, and rhetorical question. Also, there are many tweets with multiple labels assigned to them. The distribution of tweets over their labels are shown in Table 1.

Several sarcasm detection datasets are often annotated by a person who is not the author of a piece of text. However, in this dataset, the sarcastic label of each tweet is marked by its author. This makes the dataset unique in terms of capturing the sarcastic intention of the tweet-author(s). Additionally, for each sarcastic tweet, the organizers have asked the authors to rephrase the tweet-text to convey the same message without using sarcasm. However, the organizers have relied on linguistic experts to annotate the sub-categories. Experts referred Leggitt and Gibbs (2000) for the definitions of sub-categories.

Labels	**|N|**
Non-sarcastic	2601
Only-sarcasm	568
Only-irony	122
Sarcasm and irony	1
Sarcasm and satire	21
Sarcasm and overstatement	31
Sarcasm and understatement	6
Sarcasm and rhetorical questions	86
Irony and satire	4
Irony and overstatement	9
Irony and understatement	4
Irony and rhetorical question	15
Understatement and rhetorical question	2
Irony, understatement and rhetorical question	1
Sarcasm, understatement and rhetorical question	1
Total	***3468***

Table 1: Label-wise distribution of tweets. |N| refers to the sample size

3.2 Task details:

Based on the dataset they have released, the task organizers have formulated three challenges as sub-tasks. The details of the sub-tasks are,

- **Sub-task A:** Given a text, determine whether it is sarcastic or non-sarcastic;
- **Sub-task B:** This sub-task is designed for particularly English dataset. It is a binary multi-label classification task. Here, given a text, we have to determine which ironic speech category it belongs to, if any;
- **Sub-task C:** Given a sarcastic text and its non-sarcastic rephrase, i.e. two texts that convey the same meaning, determine which of the two is the sarcastic.

For all of the three sub-tasks, the organizers have informed us that precision, recall, accuracy, and macro-F1 of the participating models will be reported. According to them the main metrics of evaluation for the sub-task A is the F1-score for the sarcastic class. Similarly, for sub-task B and sub-task C, it is the macro-F1 score and the accuracy, respectively.

4 System overview

4.1 Additional resources

As shown in Table 1, the dataset is highly unbalanced, and the sample size is small. To mitigate this issue, we considered additional publicly available datasets published earlier for a similar task with some synthetically generated text. The details of such datasets are following,

- **SemEval-2018 task 3:** We used the training and test data provided by SemEval-2018 task on irony detection (Van Hee et al., 2018) as an additional resource for training of our models.

- **MUStARD:** We used the textual part of the multi-modal sarcasm detection dataset provided by Castro et al. (2019b) as an additional resource for training of our models.

- **FigLang 2020 Sarcasm:** We used the sarcasm dataset[2] released as a part of shared task of FigLang2020 [3] workshop as additional data for training of our models.

  Augmentation: For increasing the instances labeled with sub-categories in the train data provided by the task organizers, we performed data augmentation using the python NLPAUG library[4]. NLPAUG applies a set of transformations to textual datasets in order to create augmented data for deep learning models that rely on high volumes of data. We took sarcastic tweets given by organizers used the word-replacement procedure provided by NLPAUG to synthesize three additional tweets from each input tweet. We used 'distilbert-base-uncased'[5] contextual embedding as the input embedding for this process.

[2]https://github.com/EducationalTestingService/sarcasm/releases

[3]https://sites.google.com/view/figlang2020/home

[4]https://nlpaug.readthedocs.io/en/latest/overview/overview.html

[5]https://huggingface.co/distilbert-base-uncased

Source	|N|
SemEval-2022 Task 6	3468
SemEval-2018 Task 3 training	3398
SemEval-2018 Task 3 test	780
MUStARD	690
FigLang20 Sarcasm	9400
Data Augmentation (867x3)	2601

Table 2: Basic statistics of additional sources. |N| refers to the sample size.

The statistical distribution of additional resources is shown in Table 2. After elimination of the duplicates, the final dataset had 19986 tweets.

4.2 Data preprocessing

We followed the following preprocesing steps for every instance in our dataset.

- **Case conversion:** We converted the dataset into lowercase except for those words in which the whole word is in uppercase.

- **Stop-word removal:** We removed all stop-words as they contain low information. We did this using python NLTK library [6].

- **Data cleaning:** We did basic data cleaning which include removal of links, punctuation marks, floating point(.) characters and username. However, we did not apply stemming and lemmatization techniques because we believe they will distort the meaning of instances.

- **Special Tokens:** We added special tokens at the starting and ending of the instances as required by different tokenizers for respective transformer based models.

4.3 Model description

We relied on Transformer-based architectures to design our models for all sub-tasks since Transformers are regarded as state-of-the-art in NLP. We built our models using Hugging Face's Transformer library. They support generic transformer based architectures with the ability to seamlessly initialize the tokens with different pre-trained embeddings.

- **Sub-task A:** For this sub-task, we deployed the binary classifier versions of different transformer based architectures provided by the

[6]https://www.nltk.org/

Hugging Face transformer library. We particularly experimented with BERT(Devlin et al., 2019), RoBERTa(Liu et al., 2019), XLNet(Yang et al., 2019) and DistilBERT(Sanh et al., 2019) architectures. Apart from initializing the tokens with respective pre-trained embeddings, we fine-tuned the last layers of the models according to our training data. Further, we added the non-sarcastic versions of 867 sarcastic tweets given by the organizers to the training set.

- **Sub-task B:** Here, instead of using a multi-label classifier, we used six binary classifier versions of the transformer based architectures provided by the same Hugging Face library. As in the previous sub-task, here we have also experimented with BERT, RoBERTa, XLNet, and DistilBERT architectures. We constructed training data(refer Table 6) for label-wise models to fine-tune it. We merged the predictions of all six models to get the final prediction labels.
- **Sub-task C:** We formulated this sub-task as a parallel combination of two sub-task A models. We considered the same architecture for both parallel sub-components in all of our experiments.

5 Experiments and results

5.1 Sub-task A

For this sub-task, in addition to the data given by the organizers, we considered the other datasets as mentioned in Table 2. We applied the pre-processing(details in section 4.2) steps before sending them to the respective tokenizers of the considered architectures. The tokens are initialized by respective pre-trained embeddings. The dataset is divided into three parts i.e. training, validation and test set with a ratio of 0.7:0.2:0.1 for the parameter and hyper-parameter tuning. We generated the predictions for the unlabeled data provided by the organizers and submitted them in the codalab submission site[7] for evaluation. As the organizers have later released the labels for their evaluation data, we can compare all our models by ourselves too. The performance of our models for the evaluation data is reported in Table 3. The number of epochs for which the models are trained are different for different models. We trained until our models started over-fitting. Note that the best performing result reported in the table is different from that we have submitted in the codalab site. We experimented with our models even after the evaluation period and the results in Table 3 show the best performance we have achieved till date. The models submitted as a part of competition is reported in Table 5. The hyper-parameters for all models are reported in Table 4.

[7]https://codalab.lisn.upsaclay.fr/competitions/1340

5.2 Sub-task B

As stated in the previous section, for this sub-task we considered separate binary classifiers for each label. The sample size of the datasets created for individual classifiers are shown in Table 6. Note that we did not include the non-sarcastic tweets provided by the organizers in our new datasets. Rather, we added the synthetic tweets generated by the nlpaug library (see section 4.1) to amplify the label for which it is created. We didn't generate any additional synthetic text for the dataset corresponds to 'sarcasm' label as it is the dominant class in the provided 'sarcastic' data. Thus, the dataset created for the binary classification of 'sarcasm' class has the original 867 sarcasic tweets. In other label-specific datasets we increased the corresponding label tweets with the help of nlpaug library. As stated in section 4.1, we did this by generating three similar texts for each tweet tagged with the considered label. Thus, the newly created label-specific datasets have different sample size as shown in Table 6. We fine-tuned BERT model for each category of ironic speech. The organizers have evaluated sub-task B based on macro-F1 score. The results are reported in Table 7.

5.3 Sub-task C

As reported in Section 4.3, we formulated the task as two parallel combination of sub-task A models. The considered same model across the parallel sub-components. The accuracy(evaluating measure considered by the organizers) of different models are reported in Table 8. As we can infer, BERT based fine tuned model performed best on the evaluation data among all with an accuracy of **0.62**.

6 Conclusion

In this paper, we discussed our models for different sub-tasks proposed by SemEval 2022 task 6 organizers for the English dataset. We experimented

Model	Precision	Recall	F-1 score	F-1 sarcastic	accuracy
DistilBERT	0.57	0.64	0.53	0.34	0.62
XlNet	0.63	0.61	0.62	0.34	0.82
BERT	0.60	0.69	0.60	0.39	0.70
RoBERTa	***0.70***	0.68	***0.69***	***0.47***	***0.86***

Table 3: Performance measures of our models submitted for sub-task A

Model	Learning-rate	MAX_SEQ_LEN	BATCH_SIZE	EPOCHS
RoBERTa	2e-6	256	16	10
BERT	2e-5	128	32	3
Xlnet	2e-5	128	32	4
DistilBERT	5e-5	1213	16	5

Table 4: Hyper-parameters of our models.

SubTask A	
Model	*F1*
BERT	0.34
SubTask B	
Model	*Macro-F1*
BERT	0.0751
SubTask C	
Model	*Accuracy*
BERT	0.62

Table 5: Performance of our submitted models for the three tasks.

with Transformer architectures with different pre-trained language models in our submitted systems.

With our submitted models, our team "IISERB Brains", obtained the **19th rank** out of **43 teams** on sub-task A, **8th rank** out of **22 teams** on sub-task B and **13th rank** out of 16 teams in sub-task C. All of our code and links to considered data are uploaded in GitHub[8] for reproduciblity.

References

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.

Srijan Bansal, G Vishal, Ayush Suhane, Jasabanta Patro, and Animesh Mukherjee. Code-switching patterns can be an effective route to improve performance of downstream nlp applications: A case study of humour, sarcasm and hate speech detection.

Santiago Castro, Devamanyu Hazarika, Verónica Pérez-Rosas, Roger Zimmermann, Rada Mihalcea, and Soujanya Poria. 2019a. Towards multimodal sarcasm detection (an _obviously_ perfect paper). *arXiv preprint arXiv:1906.01815*.

Santiago Castro, Devamanyu Hazarika, Verónica Pérez-Rosas, Roger Zimmermann, Rada Mihalcea, and Soujanya Poria. 2019b. Towards multimodal sarcasm detection (an _Obviously_ perfect paper). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4619–4629, Florence, Italy. Association for Computational Linguistics.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Coling 2010: Posters*, pages 241–249.

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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Debanjan Ghosh, Alexander R Fabbri, and Smaranda Muresan. 2018. Sarcasm analysis using conversation context. *Computational Linguistics*, 44(4):755–792.

Debanjan Ghosh, Avijit Vajpayee, and Smaranda Muresan. 2020. A report on the 2020 sarcasm detection shared task. *arXiv preprint arXiv:2005.05814*.

Kannan Govindan, TC Edwin Cheng, Nishikant Mishra, and Nagesh Shukla. 2018. Big data analytics and application for logistics and supply chain management.

[8]https://github.com/manojmahan/iSarcasmEval-Intended-Sarcasm-Detection-In-English-main

Source	|S|	|I|	|O|	|Rq|	|St|	|U|
SemEval-2022 Task 6	867	867	867	867	867	867
Data Augmentation	0	465	120	303	75	30
Total	*867*	*1332*	*906*	*1170*	*942*	*98*

Table 6: Dataset sources used for sub-task B. Sample size for categories sarcasm, irony, overstatement, rhetorical question, satire and understatement are represented by |S|, |I|, |O|, |Rq|, |St|, |U| respectively.

Results	BERT
Macro F-1	0.075
F1-Sarcasm	0.23
F1-irony	0.096
F1-satire	0.083
F1-understatement	0.00
F1-overstatement	0.00
F1-rhetorical question	0.0414

Table 7: Macro-F1 and class-wise F1 scores for sub-task B

Model	Accuracy
RoBERTa	0.47
XlNet	0.49
DistilBERT	0.57
BERT	***0.62***

Table 8: Accuracy of our models on evaluation data provided for sub-task C.

Nikhil Jaiswal. 2020. Neural sarcasm detection using conversation context. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 77–82.

Aditya Joshi, Pushpak Bhattacharyya, and Mark J Carman. 2017. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):1–22.

Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 2015a. 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)*, pages 757–762.

Ankur Joshi, Saket Kale, Satish Chandel, and D Kumar Pal. 2015b. Likert scale: Explored and explained. *British journal of applied science & technology*, 7(4):396.

John S Leggitt and Raymond W Gibbs. 2000. Emotional reactions to verbal irony. *Discourse processes*, 29(1):1–24.

Yaochen Liu, Yazhou Zhang, Qiuchi Li, Benyou Wang, and Dawei Song. 2021. What does your smile mean? jointly detecting multi-modal sarcasm and sentiment using quantum probability. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 871–880.

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*.

Jasabanta Patro, Srijan Bansal, and Animesh Mukherjee. 2019. A deep-learning framework to detect sarcasm targets. 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)*, pages 6336–6342, Hong Kong, China. Association for Computational Linguistics.

Luke C Pilling, Chia-Ling Kuo, Kamil Sicinski, Jone Tamosauskaite, George A Kuchel, Lorna W Harries, Pamela Herd, Robert Wallace, Luigi Ferrucci, and David Melzer. 2017. Human longevity: 25 genetic loci associated in 389,166 uk biobank participants. *Aging (Albany NY)*, 9(12):2504.

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.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Himani Srivastava, Vaibhav Varshney, Surabhi Kumari, and Saurabh Srivastava. 2020. A novel hierarchical bert architecture for sarcasm detection. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 93–97.

Ishan Tarunesh, Somak Aditya, and Monojit Choudhury. 2021. Trusting roberta over bert: Insights from checklisting the natural language inference task. *arXiv preprint arXiv:2107.07229*.

Yi Tay, Luu Anh Tuan, Siu Cheung Hui, and Jian Su. 2018. Reasoning with sarcasm by reading in-between. *arXiv preprint arXiv:1805.02856*.

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.

Tony Veale and Yanfen Hao. 2010. Detecting ironic intent in creative comparisons. In *ECAI 2010*, pages 765–770. IOS Press.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.