

GetSmartMSEC at SemEval-2022 Task 6: Sarcasm Detection using Contextual Word Embedding with Gaussian Model for Irony Type Identification

Diksha Krishnan and C.Jerin Mahibha

Meenakshi Sundararajan Engineering
College, Chennai

dikshakrish5@gmail.com

jerinmahibha@gmail.com

Durairaj Thenmozhi

Sri Sivasubramaniya Nadar
College of Engineering , Chennai

theni_d@ssn.edu.in

Abstract

Sarcasm refers to the use of words that have different literal and intended meanings. It represents the usage of words that are opposite of what is literally said, especially in order to insult, mock, criticise or irritate someone. These types of statements may be funny or amusing to others but may hurt or annoy the person towards whom it is intended. Identification of sarcastic phrases from social media posts finds its application in different domains like sentiment analysis, opinion mining, author profiling and harassment detection. We have proposed a model for the shared task iSarcasmEval - Intended Sarcasm Detection in English and Arabic by SemEval-2022 considering the language English. The Subtask A and Subtask C were implemented using a Convolutional Neural Network based classifier which makes use of ELMo embeddings. The Subtask B was implemented using Gaussian Naive Bayes classifier by extracting TF-IDF vectors. The proposed models resulted in macro-F1 scores of 0.2012, 0.0387 and 0.2794 for sarcastic texts in Subtasks A, B and C respectively.

1 Introduction

In the Internet era, specifying user comments, views and opinions through social media has become very common and these may not be specified directly and can include indirect phrases with implicit meanings (Abu Farha and Magdy, 2020). The posts may also represent undesirable characteristics using positive words and they may not be formal in nature. It is common to use abbreviations, uncommon, ambiguous and multilingual words in social media posts and no predefined structures are explicitly defined for sarcastic messages (Sarsam et al., 2020). So the identification of sarcastic posts cannot be considered as a direct process. Exhaustive understanding of the contextual meaning of the posts is considered as an important factor in identifying sarcastic posts. Sarcasm detection is considered as a challenging task associated with sentiment

analysis. Sarcasm detection also plays an important role in analysing the voice of the customer based on which major decisions will be taken.

Sarcasm requires some shared knowledge between speaker and audience and it is considered as a profoundly contextual phenomenon. Using sarcastic phrases in sentences is a common way of ironic or satirical speech for the common man. That being said, social media platforms, such as Twitter, Facebook and YouTube contain millions of tweets and comments that include sarcastic phrases, thus making it a field of study under the domain of NLP. For example, a user can change a supposedly negative comment using positive words as in the sentence, “*It is awesome to go to bed at 3 am #not*”. In such cases it becomes important to ensure that the right sentiment is drawn out of the sentence through proper analysis (Vu et al., 2018).

The shared task iSarcasmEval was part of SemEval 2022 (Abu Farha et al., 2022) and there were three subtasks associated with it. Two of the subtasks - Subtask A and Subtask B, were conducted for both English and Arabic languages, while Subtask C only pertains to English. We as a team participated in all the three subtasks associated with the language English.

Subtask A: Sarcastic vs. Non-Sarcastic

The first subtask was a straightforward Binary Classification problem, in which the model had to predict whether a given phrase is sarcastic or not. For example, “*I work 40 hours a week for me to be this poor.*” is sarcastic, whereas “*Her husband is serving a three-year sentence for fraud.*” is non-sarcastic.

Subtask B: Different types of irony

The second subtask was a Multi-Label Classification problem, where the aim was to identify which type of irony a given sentence falls under from six specific labels: sarcasm, irony, satire, understatement, overstatement and rhetorical question.

Subtask C: Differentiation between sarcastic

phrase and its rephrase

The final subtask is another Binary Classification problem, wherein, given a sarcastic phrase and its rephrase, the model needs to identify which of the two is sarcastic. For example, the phrase “*Taxes are just the best and I cannot wait to pay more*” is sarcastic, whereas its rephrase - “*I dislike paying taxes.*” is non-sarcastic (Goutte et al., 2014).

Considering all the three subtasks the training of the proposed model was done using the training data set provided for the corresponding task and language. This model was then tested with a testing data set provided by the shared task, based on which the task was evaluated.

2 Related Works

Sarcasm detection can be posed as a text classification task in which the given text needs to be identified as sarcastic or not. In simple systems it could be considered as a binary classification task where the presence of sarcasm is only detected. But if the irony associated with the sarcastic message needs to be identified then it is considered as a multi label classification task. Different machine learning and deep learning models could be used for identifying sarcasm in text and it is not evident that a particular algorithm provides the best result for any data. The features of the data set like the number of instances in the data set, distribution of data in the training data set are important factors on which the performance of the algorithm relies. So it becomes necessary to analyse the data set, to choose the model for implementing the classification task.

It had been shown that Support Vector Machine provided the best performance for sarcasm detection considering posts from twitter (Sarsam et al., 2020). A combination of Convolutional Neural Network (CNN) and SVM had also offered high prediction accuracy. Even though feature rich SVM model perform well, Avinash Kumar and et al. (Kumar et al., 2020) had shown better performance by combining multi-head attention mechanism with bidirectional long-short memory (BiLSTM) to detect sarcastic messages.

Use of deep learning models with BiLSTM to detect sarcasm from Arabic texts had been illustrated by (Abu Farha and Magdy, 2020). A sarcasm detector had been implemented by fine tuning BERT and considering both affective and contextual features (Babanejad et al., 2020). The fea-

tures of sarcastic text had been captured using a multi-level memory network to take care of sentiment semantics and the contrast between it and the situation in each sentence (Ren et al., 2020). Different transformer-based language models were used for sarcasm detection in Arabic language text and had found that MARBERT and AraELECTRA performed well and AraGPT2 had shown poor performance (Abu Farha and Magdy, 2021).

Pre-trained transformer models when ensembled with Recurrent Convolutional Neural Network to form hybrid neural architecture to detect sarcastic messages had resulted in better performance when compared to other relevant state-of-the-art methodologies (Potamias et al., 2020). A hybrid model implemented using bidirectional long short-term memory with a softmax attention layer and convolution neural network when used for sarcasm detection had resulted in better performance (Jain et al., 2020). Sarcasm detection had been implemented by taking into account the contextual information using a dual channel Convolutional Neural Network and the user’s expression habit had been identified using attention mechanisms (Du et al., 2022).

In addition to the above approaches, various novel approaches like statistical approaches, graph based approaches, fuzzy logic based approaches and pseudo labelling approaches had been used for identifying sarcastic messages. A complex-valued fuzzy network had been used to identify the text with sarcasm by leveraging the mathematical formalisms of quantum theory and fuzzy logic (Zhang et al., 2021). Sarcasm detection had been implemented by taking the contextual information of a sentence into account in a sequential manner using the concept of pseudo-labeling (Kumar Jena et al., 2020), (Kalaivani and Thenmozhi, 2020). Long-range literal sentiment inconsistencies had been taken into account in sarcasm detection by constructing an affective graph and a dependency graph for each sentence and had then used an Affective Dependency Graph Convolutional Network (ADGCN) framework for the classification process (Lou et al., 2021). Statistical approach had been proposed for sarcasm detection by combining TF-IDF features with the important features related to sentiments and punctuations that are identified using chi-square test (Gupta et al., 2020).

Social media posts may have both text and images associated with it and identifying sarcasm in

Task	Category	Instances
Subtask A	Sarcastic	867
	Non Sarcastic	2601
Subtask B	Sarcasm	713
	Irony	155
	Satire	25
	Under Statement	10
	Over Statement	40
	Rhetorical question	101
Subtask C	Rephrase	867

Table 1: Data Distribution

such case had been implemented using a multi-modal framework using Coupled-Attention Networks (CANs) which captures and integrates information from both text and image for the classification task (Zhao et al., 2021). As the imbalanced nature of the data set affects the performance of the model oversampling had been done to convert the data set into a balanced one and had shown that SMOTE and BorderlineSMOTE-1 techniques when used for oversampling had resulted in the improvement of performance (Banerjee et al., 2020). It could be summarized from the related works that identifying sarcasm from social media posts is an emerging research area that requires more insights. Different techniques like traditional machine learning models such as SVM and Random Forest, Convolutional Neural Network based models, Transformer models and Ensemble models could be used for this purpose. It could be found that the performance of the approach depends on the dataset on which the model is being trained (Abdulraheem et al., 2015). It is hard to identify a particular approach that can detect sarcastic messages under any circumstances which provides the best performance.

3 Data set

The data set that we used to implement sarcasm detection was the training and the test dataset that was provided by the organisers of the shared task. Each instance of the training dataset had the following informations attached to it:

1. Label specifying the sarcastic nature of the text
2. Rephrase text that convey the same message of the text non-sarcastically

3. Label specifying the category of ironic speech which includes sarcasm, irony, satire, understatement, overstatement and rhetorical question

There were 3468 instances in the training data set of which 867 instances were under the sarcastic category and remaining 2601 instances were under the non sarcastic category. This shows the unbalanced nature of the data set. The test data had 1400 instances for which the predictions had to be done using the proposed model. The distribution of the data in the training dataset is shown in Table 1. As separate data set was not provided for the evaluation purpose, under both category 80% of the training data instances were used for training purpose and 20% of the instances from the training data set was used for the evaluation purpose.

4 System Description

The proposed methodology uses ELMo embedding based Convolutional Neural Network model for implementing the Subtasks A and C with an embedding layer followed by two dense layers. The Subtask B has been implemented using TF-IDF based Gaussian Naive Bayes classifier.

4.1 ELMo Model

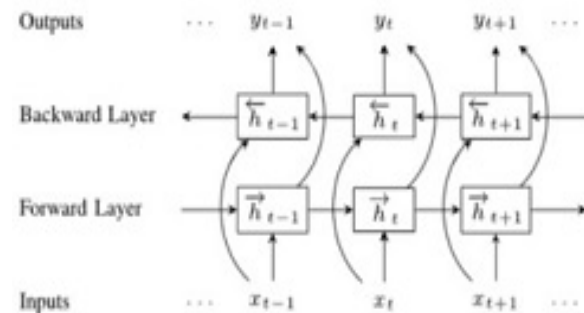


Figure 1: Architecture of ELMo Model

It stands for Embeddings from Language Models and is a novel way to represent words in vectors or embeddings (Peters et al., 2018). In ELMo the syntax and semantics of the word and the linguistic context associated with it are modelled as a deep contextualised word representation. Huge text corpus had been used to pretrain the model and is constructed using deep bidirectional models¹. It is implemented with 4096 units and the input embedding transform using 2048 convolutional filters.

¹<https://allennlp.org/allennlp/software/elmo>

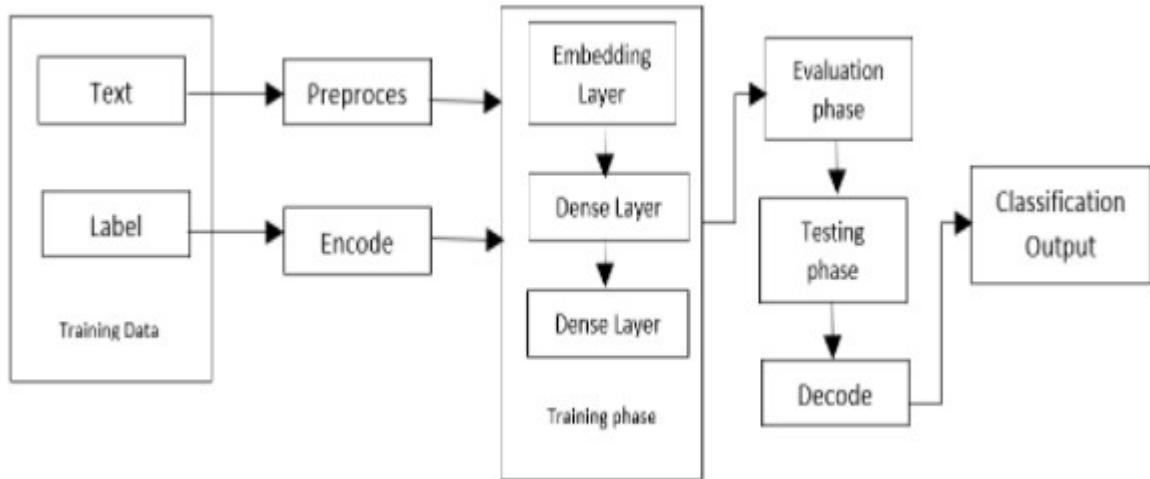


Figure 2: Proposed Architecture

The context of the word usage forms the base for the word representation. ELMo word representations take the entire input sentence into equation for calculating the word embeddings. The architecture of ELMo model² is shown in Figure 1.

ELMo is a bidirectional language model. The forward LM is a deep LSTM that goes over the sequence from start to end to predict the token and the backward LM is a deep LSTM that goes over the sequence from end to start to predict the token.

$$P(x_i | y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Figure 3: Likelihood of Gaussian Naive Bayes

4.2 Gaussian Naive Bayes Classifier

It follows Gaussian Normal Distribution and is a variant of Naive Bayes which are a group of supervised machine learning classification algorithms based on the Bayes theorem. The concept of conditional probability is used while using this for the classification problem³. The likelihood of the features in Gaussian Naive Bayes Classifier is represented by Figure 3. Predictions using this classifier is done by providing new input values for the parameters which will result in an estimated

probability by the Gaussian function.

The task of determining whether the given text is sarcastic or non-sarcastic represents a binary classification problem, which is Subtask A. The task of determining the sarcastic one from two texts that convey the same meaning also represents a text classification problem which is the problem statement associated with Subtask C. The proposed model uses an ELMo model to implement the above two tasks. Subtask B is a binary multi-label classification task in which the correct ironic speech category has to be identified from the given set of labels. Figure 2 shows the architecture of the proposed model.

The first step associated with all the subtasks is to prepare the data set, which involves preprocessing of the text within. This is carried out by removing any escape sequences and stop words associated with the text, generating the associated tokens and lemmatizing the same. The idea behind preprocessing is to remove the parts from the text which do not contribute to the actual intent of the text.

For implementing Subtask A and C which were binary classification problem, the labels were normalized by using a one hot encoding scheme, which transformed the labels into a categorical value for which embedding is done and the encoded labels are returned⁴. The encoded data is used to train the model which is a Convolutional Neural Network with an embedding layer followed by two dense layers. The embedding layer gener-

²<https://andy-nguyen.medium.com/create-a-strong-text-classification-with-the-help-from-elmo-e90809ba29da>

³<https://jakevdp.github.io/PythonDataScienceHandbook>

⁴<https://scikit-learn.org/stable/modules/generated>

Metric	Score
F1-Sarcastic	0.2012
F-Score	0.5101
Precision	0.5137
Recall	0.5196
Accuracy	0.705

Table 2: Subtask A Scores

ates the ELMo embeddings which is implemented with the help of tensorflow hub. The first dense layer has 256 units and makes use of relu activation function. The output is generated from the second dense layer with two units as it performs Binary Classification. The activation function used by the last layer is softmax. As Subtask B involved multi label classification, TF-IDF vectors were generated for the input text and the classification was carried out using a Gaussian Naive Bayes classifier⁵, which uses the concept of Bayes theorem.

The training data set provided as a part of the shared task was used for training the model. As a separate data set was not provided for validation, 20% of the training instances were selected and used for the evaluation process. Finally the testing phase was implemented using the testing data set provided for the shared task.

5 Results

The metrics that were considered for the evaluation of all the three subtasks were precision, recall, accuracy and macro-F1 score. Precision represents the ratio of the number of correct positive results to the number of positive results predicted by the classifier. The recall measures the model’s ability to detect positive samples. The higher the recall, the more positive samples are detected. Classification accuracy is the ratio of number of correct predictions to the total number of input samples. F1 score is an overall measure of a model’s accuracy that combines precision and recall. A high F1 score means that the classification has resulted with low number of false positives and low false negatives. The proposed model resulted in an F1 score for sarcastic texts as 0.2012 based on which Subtask A was evaluated and we were ranked 36 on the leaderboard. Table 2 shows the values that were obtained for various metrics like precision, accuracy and recall considering Subtask A.

Subtask B was evaluated based on the macro-F1

⁵https://scikit-learn.org/stable/modules/naive_bayes.html

Metric	Score
Macro F	0.0387
F1-Sarcasm	0.2321
F1-Irony	0.0000
F1-Satire	0.0000
F1-Understatement	0.0000
F1-Overstatement	0.0000
F1-Rhetorical Question	0.0000

Table 3: Subtask B Scores

Metric	Score
Accuracy	0.3400
F-Score	0.2794

Table 4: Subtask C Scores

score of the model and the proposed model had resulted in a macro-F1 score of 0.0387. We ranked 19 on the leaderboard under this category. Table 3 shows the F1 scores that were obtained for different type of sarcastic texts like Irony, Satire, Understatement, Overstatement and Rhetorical Question.

Accuracy was the metric that was used for the evaluation of Subtask C, and our model achieved an accuracy of 0.34 with rank 15 on the leaderboard. The F1 score that we obtained for this subtask was 0.2794 and these are tabulated in Table 4.

6 Conclusions

Sarcasm detection has become an important area of research as it is interlinked with different areas of application that includes sentiment analysis, opinion mining, offensive and hate speech detection. Having this in mind SemEval-2022 had come up with the task of Sarcasm detection which was represented by three subtasks namely sarcasm detection, identifying the type of irony associated with the sarcastic text and identifying whether the text or its rephrase is sarcastic. We have applied ELMo embedding based Convolutional Neural Network model for implementing the binary Subtasks A and C. Gaussian Naive Bayes classifier based on TF-IDF vectors was used to implement Subtask B. All the three subtasks were implemented considering the language English. The performance of the models used, were not up to the mark and it is found from the task overview that the transformer models when applied over this tasks provides better results.

Dataset for sarcasm detection could be created with contextual information which can help in ef-

fectively detecting sarcasm. Usage of hybrid approaches where different machine learning and deep learning models are combined can also facilitate efficient detection of sarcasm from text. Often it could be observed that sarcasm is not in the text, but could be detected from the intonation or facial expression, which has made multimedial sarcasm detection also as a promising research area.

References

- Ajiboye Abdulraheem, Ruzaini Abdullah Arshah, and Hongwu Qin. 2015. [Evaluating the effect of dataset size on predictive model using supervised learning technique](#). *International Journal of Software Engineering Computer Sciences (IJSECS)*, 1:75–84.
- Ibrahim Abu Farha and Walid Magdy. 2020. [From Arabic sentiment analysis to sarcasm detection: The ArSarcasm dataset](#). In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 32–39, Marseille, France. European Language Resource Association.
- Ibrahim Abu Farha and Walid Magdy. 2021. [Benchmarking transformer-based language models for Arabic sentiment and sarcasm detection](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 21–31, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- 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.
- Nastaran Babanejad, Heidar Davoudi, Aijun An, and Manos Papagelis. 2020. [Affective and contextual embedding for sarcasm detection](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 225–243, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Arghasree Banerjee, Mayukh Bhattacharjee, Kushankur Ghosh, and Sankhadeep Chatterjee. 2020. [Synthetic minority oversampling in addressing imbalanced sarcasm detection in social media](#). *Multimedia Tools and Applications*, 79(47):35995–36031.
- Yu Du, Tong Li, Muhammad Salman Pathan, Hailay Kidu Teklehaimanot, and Zhen Yang. 2022. [An effective sarcasm detection approach based on sentimental context and individual expression habits](#). *Cognitive Computation*, 14(1):78–90.
- Cyril Goutte, Michel Simard, and Marine Carpuat. 2014. [CNRC-TMT: Second language writing assistant system description](#). In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 192–197, Dublin, Ireland. Association for Computational Linguistics.
- Rahul Gupta, Jitendra Kumar, Harsh Agrawal, and Kunal. 2020. [A statistical approach for sarcasm detection using twitter data](#). In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 633–638.
- Deepak Jain, Akshi Kumar, and Geetanjali Garg. 2020. [Sarcasm detection in mash-up language using soft-attention based bi-directional lstm and feature-rich cnn](#). *Applied Soft Computing*, 91:106198.
- A Kalaivani and D Thenmozhi. 2020. [Sarcasm identification and detection in conversation context using bert](#). In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 72–76.
- Avinash Kumar, Vishnu Teja Narapareddy, Veerubhotla Aditya Srikanth, Aruna Malapati, and Lalita Bhanu Murthy Neti. 2020. [Sarcasm detection using multi-head attention based bidirectional lstm](#). *IEEE Access*, 8:6388–6397.
- Amit Kumar Jena, Aman Sinha, and Rohit Agarwal. 2020. [C-net: Contextual network for sarcasm detection](#). In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 61–66, Online. Association for Computational Linguistics.
- Chenwei Lou, Bin Liang, Lin Gui, Yulan He, Yixue Dang, and Ruifeng Xu. 2021. [Affective Dependency Graph for Sarcasm Detection](#), page 1844–1849. Association for Computing Machinery, New York, NY, USA.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Rolandos Alexandros Potamias, Georgios Siolas, and Andreas . Georgios Stafylopatis. 2020. [A transformer-based approach to irony and sarcasm detection](#). *Neural Computing and Applications*, 32(23):17309–17320.
- Lu Ren, Bo Xu, Hongfei Lin, Xikai Liu, and Liang Yang. 2020. [Sarcasm detection with sentiment semantics enhanced multi-level memory network](#). *Neurocomputing*, 401:320–326.
- Samer Muthana Sarsam, Hosam Al-Samarraie, Ahmed Ibrahim Alzahrani, and Bianca Wright. 2020. [Sarcasm detection using machine learning algorithms in twitter: A systematic review](#). *International Journal of Market Research*, 62(5):578–598.

- Thanh Vu, Dat Quoc Nguyen, Xuan-Son Vu, Dai Quoc Nguyen, Michael Catt, and Michael Trenell. 2018. [Nihrio at semeval-2018 task 3: A simple and accurate neural network model for irony detection in twitter.](#)
- Yazhou Zhang, Yaochen Liu, Qiuchi Li, Prayag Tiwari, Benyou Wang, Yuhua Li, Hari Mohan Pandey, Peng Zhang, and Dawei Song. 2021. [Cfn: A complex-valued fuzzy network for sarcasm detection in conversations.](#) *IEEE Transactions on Fuzzy Systems*, 29(12):3696–3710.
- Xuan Zhao, Jimmy Huang, and Haitian Yang. 2021. [Cans: Coupled-attention networks for sarcasm detection on social media.](#) In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.