Sarcasm Detection in News Headlines using Evidential Deep Learning-based LSTM and GRU

Md. Shamsul Rayhan Chy[1*], Fahim Faisal Rafi[1],
Md. Shamsul Rahat Chy[1], Md. Farhadul Islam[1],
Md Sabbir Hossain[1], Annajiat Alim Rasel[1]

*Corresponding author(s). E-mail(s):
md.shamsul.rayhan.chy@g.bracu.ac.bd;
Contributing authors: fahim.faisal.rafi@g.bracu.ac.bd;
shamsul.rahat.chy@g.bracu.ac.bd; md.farhadul.islam@g.bracu.ac.bd;
md.sabbir.hossain1@g.bracu.ac.bd; annajiat@gmail.com;

Abstract

Sarcasm has become quite inter-related with the day to day life of all. In news robust sarcasm is often used to grab the attention of the viewers. This research aims to detect sarcasm using Evidential deep learning. This technique uses uncertainty estimaions for identifying the sentiments from news headlines dataset. Also, LSTM and GRU have been used with Evidential deep learning approach. The purpose of using LSTM is that it can classify texts from headlines in order to analysis the sentiments. Moreover, we have used GRU which is an recurrent neural networks (RNN) and it effectively models sequential data. The architecture of the GRU network is ideally suited for identifying dependencies and extended contextual relationships within news headings. Overall, our proposed model uses Evidential deep learning based LSTM and GRU to identify the sentiments of robust sarcasms from news headlines.

Keywords: evidential deep learning, lstm, gru, sarcasm detection, nlp

1 Introduction

Sarcasm is used to express feeling in a humouric way where positive words are generally used to make a statement but it contains a negative meaning in the written text. Using sarcasm is a technique to grab attention of others as this type of statements are often very facetious. Sarcasm is also used to express an opinion on a particular viewpoint. Its recognition has becoming more crucial in many applications of natural language processing. In this study, we discuss methods, problems, difficulties, and potential future applications of sarcasm detection [1]. In this research, we have used Evidential deep learning technique to analize the sentiment of news headlines. Evidential deep learning entails the incorporation of uncertainty estimation into deep learning models through the utilisation of Bayesian inference procedures. The model can identify circumstances when it lacks adequate data or runs across unclear inputs thanks to uncertainty estimations. Uncertainty estimation also supports with the assessment of model confidence. It is crucial to have a grasp of the dependability and trustworthiness of the model's predictions in order to be able to use it in key applications such as medical diagnosis and legal analysis. We have also integrated LSTM with Evidential deep learning and the purpose of it is to identify the sentiment of the news headlines. LSTM can process each word in news headlines in sequence and the final output of it is used to classify the sentiment of the respective headline.

2 Related Works

Robust sarcasm detection from news headlines is a challenging with the use of Evidential deep learning. There are some related works with sarcasm detection. Various kind of techniques has been used to detect the sentiment analysis of sarcasms.

In the paper [2], the authors have experimented with an algorithm of STSM which is sentiment topic sarcasm mixture. A lot of features has been used for the development of the model such as pragmatic features and lexicon based. The main concept of this model is that the influence of some topics in detecting sarcasm is more than others.

Paper [3] provides an ensemble model to detect sarcasm on the internet. The used dataset is prepared on previously trained various word-embedding models. Here weighted average obtained the highest accuracy in case of both the datasets. Authors of paper [4] proposed a model named C-Net. It extracts contextual knowledge from texts in a serial fashion way. Then it categorises the texts into sarcastic or non-sarcastic.

Authors of [5] built a transformer based method where recurrent CNN-RoBERTA model is used to find the sarcasm in given statements. In paper [6], authors have proposed a new behavioral model for detecting sarcasm. They have used a lexicon-based technique. Their SCUBA model can detect whether a person is sarcastic or not using past data. This model uses various approach and among them SCUBA++ achieves the highest accuracy 92.94%.

The authors of [7] used a set of features. The features are derived in a manner that utilises various components of the tweet and encompasses various forms of sarcasm. Their proposed approach achieves higher accuracy of 83.1% compared to some baseline approaches like n-grams. In [8], authors have used supervised machine learning methods for detecting sarcasm on Czech and English Twitter datasets. The paper focuses on the document level sarcasm detection. They have used various n-grams with frequency greater than three and a set of language independent features. SVM classifier with the feature set achieves best result on Czech dataset with F-measure 0.582.

From [9], there are two different groups of machine learning algorithms AMLA and CMLA. In AMLA group the most prevalent method was discovered to be SVM (22.58%), followed by Logistic Regression Method (19.35%), Nave Bayes (9.67%), and Random Forest. For recognising the sarcastic tweets, algorithms in the CMLA group were found to be utilised less often (3.22%). So, here the AMLA group algorithms are the ones that are most frequently used to detect sarcasm. Authors of [10] used Hybrid Ensemble Model with Fuzzy Logic to detect sarcasm over social media platforms. Here they used the Reddit dataset, twitter dataset and the headlines dataset. Their model makes use of BERT-base, Word2Vec, and GloVe. The stated portion from the three previously discussed strategies is used by the fuzzy logic layer to assess the categorization probability of all three processes. This model achieves the best accuracy in all three datasets which are respectively 85.38%, 86.8% and 90.81%.

3 Dataset

In our research, we have used the news headlines dataset [11]. The purpose of using this dataset is to avoid label and language noise. This dataset contains sarcastic News headlines from the news website named TheOnion. On the other hand, this dataset also contains non-sarcastic and real parts which has been collected from the HuffPost news website. In this dataset there are 28,619 headlines which contains both sarcastic and non-sarcastic. The language used in the headlines of this dataset is formal as all the writings have been done by professional writers. That is why there is very little chance of spelling mistake in the headlines. Also, as one of the news website that provided only the sarcastic texts, that is why there is high chance that the quality of label is controlled in the respective dataset.

4 Methodology

4.1 Evidential deep learning

Evidential deep learning is used in natural language processing (NLP) to resolve the limitations of conventional deep learning models. This technique used uncertainty estimations. The weights and biases in this model are treated as random variables with prior distributions by Bayesian neural networks (BNNs), which were used in its construction. We use variational inference, a method that improves the Evidence Lower Bound (ELBO) by repeatedly changing the variational distribution to closely resemble the real posterior distribution, to approximate the posterior distribution over the model parameters. The fluctuation seen in these forecasts offers insightful information about the level of uncertainty around the model's results. Also, confidence intervals are derived from uncertainty estimations. Instances where the model lacks assurance or runs across inputs that are intrinsically unclear are indicated by higher uncertainty levels. The model's confidence is shown by variables like predictive variance, entropy, or quantiles of the predictive distribution.

4.1.1 Theory of Evidence

Neural Networks can have K outputs and the equality of it can be written as

$$u + \sum_{k=1}^{K} b_k = 1$$

Here b_k is interpreted as the belief mass of the k^{th} class and the uncertainty mass of the particular outputs is u and b_k is defined as follows

$$b_k = \frac{e_k}{S}$$

Now the e_k is k^{th} class evidence and S is strength of Dirichlet and is defined as follows

$$S = \sum_{k=1}^{K} (e_k + 1)$$

which leaves u the following portion

$$u = \frac{K}{S}$$

Replacing $e_k + 1$ with a_k

$$\alpha_k = e_k + 1$$

Here resultant sinplex vector a is used as the density in a Dirichlet

$$D(\boldsymbol{p} \mid \boldsymbol{\alpha}) = \begin{cases} \frac{1}{B(\boldsymbol{\alpha})} \prod_{i=1}^{K} p_i^{\alpha_i - 1} & \text{for } \boldsymbol{p} \in \mathcal{S}_K \\ 0 & \text{otherwise} \end{cases}$$

$\mathcal{S}_K$ can be defined as

$$\mathcal{S}_K = \left\{ \boldsymbol{p} \mid \sum_{i=1}^{K} p_i = 1 \text{ and } 0 \leq p_1, \ldots, p_K \leq 1 \right\}$$

Finally, the probability of k^{th} is calculated as

$$\hat{p}_k = \frac{\alpha_k}{S}$$

4.1.2 Reliability Evaluation of the Classification Algorithm

Classification Algorithm Reliability Evaluation (Stable Operational Profile) According to the literature, we presume that the chance of not failing on a randomly selected input $d_r \in D$ [12] is how the black-box dependability is stated. The priors $f_i(x)$ are set to Beta $(\alpha_i = 1, \beta_i = 1)$ with the assumption that each class represents an operational profile of traffic sign recognition and that no prior information regarding the incidence of failures within partitions is known. Let's assume that N_i represents the quantity of test photos sent to the algorithm as input and r_i represents the quantity of errors.

The Dirichlet distribution $D\left(\alpha_1, \ldots, \boldsymbol{\alpha_n}\right)$, which modelled the OPP prior to the new observation, will be transformed by the new information $N_1, \ldots, N_n$ into:

$$D\left(\boldsymbol{\alpha}_1 + N_1, \ldots, \boldsymbol{\alpha}_n + N_n\right)$$

The revised distribution of the operational profile's or partition S_i's conditional probability of failing to recognise class i will be:

$$f_{F_i} = B\left(\alpha_i + r_i, \beta_i + N_i - r_i\right)$$

The expected value of f_{F_i} can be calculated as:

$$E\left[F_i\right] = \frac{\alpha_i + r_i}{\beta_i + \alpha_i + N_i}$$

Finally the reliability is calculated as taking each OPP_i as 1/10

$$E[R] = 1 - \sum_{i=1}^{43} OPP_i \times E\left[F_i\right] = 1 - 0.1 \times \sum_{i=1}^{43} \frac{\alpha_i + r_i}{\beta_i + \alpha_i + N_i}$$

4.2 LSTM

LSTM is known as Long Short Term Memory. The idea to use LSTM is that is can use to classify texts and to analysis sentiment of texts. LSTM use the mechanism of memory cell where it has three gates connected to each of the cell. Input gate, forget gate and output gate. With the use of these LSTM is capable to retain any information or to forget it. A sigmoid activation function is used to decide which data to keep or discard from input or previous cell. This technique make predictions based on the relavant data. Along with the capability to use memory cell it can make effective predictions in order to analysis the sentiment of any written text. For analyzing sentiment it also uses an embedding layer. This layer helps to map each word of a sentence and each word here gets a vector representation and after training it can learn the most informative vector representation for each word. The model then processes sequence of word vectors and a sentiment based on the prediction is achieved. It also uses backpropagation strategy for training the data. For cross checking the performance of LSTM metrices like accuracy, precision, recall and F1 score can be used.

4.3 GRU

GRU is a form of architecture for recurrent neural networks (RNN) that effectively models sequential data. GRU incorporates gating mechanisms that allow the network to capture and retain pertinent data over extended sequences. This gating mechanism, which consists of reset and update gates, permits the model to selectively update and retain pertinent information while discarding irrelevant or redundant data. The GRU's gating mechanism regulates the passage of information through the network, allowing it to capture textual dependencies and contextual information. Backpropagation through time (BPTT), which extends the concept of backpropagation to sequential data, is used to train the GRU model. This method allows for the efficient updating of the model's weights and biases by propagating error gradients over time. GRU is utilised to assess a model's ability to generalise to unobserved data by evaluating its performance on the validation and test sets. GRU permits the network to selectively update and retain pertinent data while discarding irrelevant or redundant data. This mechanism facilitates comprehension of the subtleties of sarcasm and tone in headlines.

5 Results and Analysis

In this section, we analyze the performance of our proposed models. We assess the models using a variety of performance criteria, such as test accuracy, test AUC score, and Reliability.

Model	**LSTM**	**GRU**
Number of Parameters	1,499,309	1,447,449
Test Accuracy	82%	78%
Test AUC	0.81	0.77
Reliability Score	0.49	0.43

Here, the table above shows the raw and uncertain-aware performances of LSTM and GRU. The LSTM model was trained with weight parameters of 1,499,309. During the evaluation, it attained an accuracy of 82% and an AUC-score of 0.81. The reliability score is 0.49. The GRU model, on the other hand, performed slightly worse having slightly less weight parameters of 1,447,149. It outperformed the LSTM model in terms of accuracy and AUC-score, attaining 78% and 0.77 respectively. Moreover, the model achieves an reliability score of 0.43.

The training vs validation curves are shown in Figure 1 and in Figure 2. The epochs are the same and requires only 4. The fitting is considerably more stable in GRU but the performance of LSTM is better.

Considering the reliability score and other evaluation measures, we can say that LSTM is a better model to go forward with. The small difference of weight parameters is negligible since the performance is noticeably better.

6 Conclusion & Future Works

Sarcasm is used more often to grab the attention of readers and it has now gained more importance to work with sarcasm detection in various Natural Language Processing

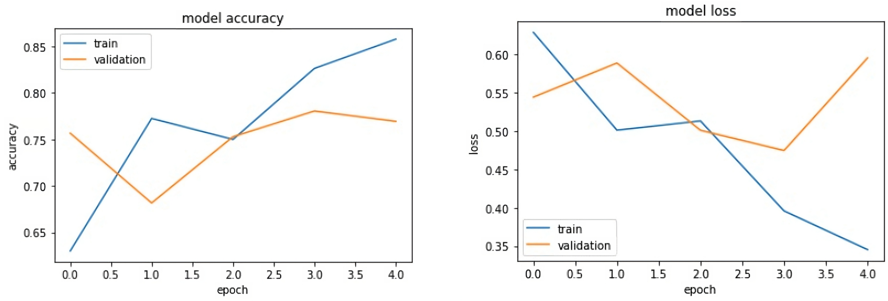

Fig. 1: Accuracy and Loss curve of LSTM Model

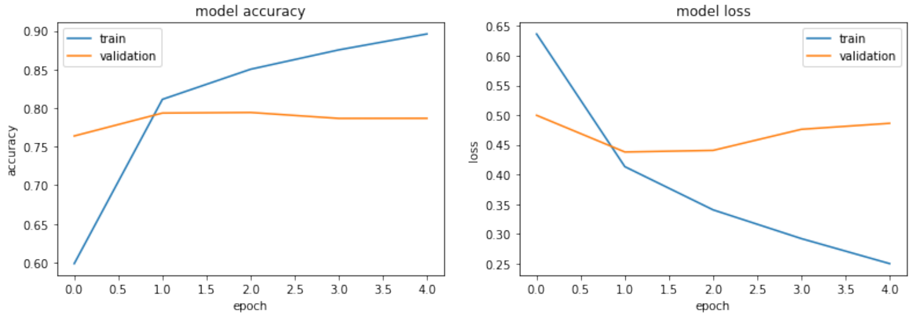

Fig. 2: Accuracy and Loss curve of GRU Model

Applications. In this paper we have worked on news headlines dataset. This dataset contains both sarcastic and non-sarcastic news headlines. By using this dataset we have trained Evidential deep learning based LSTM model and GRU model. Between these two models the LSTM based model achieves greater accuracy with 82 percent than GRU. It has also better AUC-score than the GRU model. Also, we have the readibility score and in this section also the LSTM performs the better result. We would like to extend our work with other uncertainty measuring methods such as monte carlo dropout, active learning etc in our future works. Since, detecting sarcasm is a difficult task and the ambiguity is quite high, multiple uncertainty and reliability measures should be applied.

References

[1] Chaudhari, P.P., Chandankhede, C.: Literature survey of sarcasm detection. 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2041–2046 (2017)

[2] Krishnan, N., Rethnaraj, J., Saravanan, M.: Sentiment topic sarcasm mixture model to distinguish sarcasm prevalent topics based on the sentiment bearing words in the tweets. Journal of Ambient Intelligence and Humanized Computing **12** (2021) https://doi.org/10.1007/s12652-020-02315-1

[3] Goel, P., Jain, R., Nayyar, A., Singhal, S., Srivastava, M.: Sarcasm detection using deep learning and ensemble learning. Multimedia Tools and Applications **81** (2022) https://doi.org/10.1007/s11042-022-12930-z

[4] Jena, A., Sinha, A., Agarwal, R.: C-net: Contextual network for sarcasm detection, pp. 61–66 (2020). https://doi.org/10.18653/v1/2020.figlang-1.8

[5] Potamias, R., Siolas, G., Stafylopatis, A.: A Transformer-based Approach to Irony and Sarcasm Detection

[6] Rajadesingan, A., Zafarani, R., Liu, H.: Sarcasm detection on twitter: A behavioral modeling approach. WSDM 2015 - Proceedings of the 8th ACM International Conference on Web Search and Data Mining, 97–106 (2015) https://doi.org/10.1145/2684822.2685316

[7] Bouazizi, M., Ohtsuki, T.: A pattern-based approach for sarcasm detection on twitter. IEEE Access **4**, 5477–5488 (2016)

[8] Ptácek, T., Habernal, I., Hong, J.: Sarcasm detection on czech and english twitter. In: International Conference on Computational Linguistics (2014)

[9] Sarsam, S.M., Al-Samarraie, H., Alzahrani, A.I., Wright, B.: Sarcasm detection using machine learning algorithms in twitter: A systematic review. International Journal of Market Research **62**(5), 578–598 (2020) https://doi.org/10.1177/1470785320921779 https://doi.org/10.1177/1470785320921779

[10] Sharma, D., Singh, B., Agarwal, S., Pachauri, N., Alhussan, A., Abdallah, H.: Sarcasm detection over social media platforms using hybrid ensemble model with fuzzy logic. Electronics **12**, 937 (2023) https://doi.org/10.3390/electronics12040937

[11] Misra, R., Arora, P.: Sarcasm detection using news headlines dataset. AI Open **4**, 13–18 (2023) https://doi.org/10.1016/j.aiopen.2023.01.001

[12] Pietrantuono, R., Popov, P., Russo, S.: Reliability assessment of service-based software under operational profile uncertainty. Reliability Engineering System

Safety **204**, 107193 (2020) https://doi.org/10.1016/j.ress.2020.107193