SEMARAK ILMU
PUBLISHING
202103268166(003316878-P)

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Detection of Sarcasm Using Bi-Directional RNN Based Deep Learning Model in Sentiment Analysis

Muddada Murali Krishna[1,*], Midhunchakkaravarthy[2], Jayavani Vankara[3]

[1] Department of CSE, Lincoln University College, Petaling Jaya 47301, Malaysia
[2] Faculty of Computer Science and Multimedia, Lincoln University College, Petaling Jaya 47301, Malaysia
[3] Department of CSE, GITAM School of Technology, GITAM (Deemed to be University), Visakhapatnam, Andhra Pradesh, India

ARTICLE INFO

Article history:
Received 16 April 2023
Received in revised form 3 June 2023
Accepted 11 July 2023
Available online 1 August 2023

Keywords:
Sarcasm identification; Deep Learning technique; Bi-directional LSTM; RNN

ABSTRACT

Detecting sarcasm in text can be challenging, but it's an opportunity to exercise the ability to consider various perspectives and aspects. Considering multiple viewpoints while analysing sentences can lead to comprehensive results and constructive discussions. Sentiment analysis algorithms are constantly improving and becoming more accurate in identifying sarcasm, even in implicit expressions where the sarcasm is not explicitly stated. It's exciting to have the opportunity to detect sarcasm conveyed through subtle cues, as it adds another layer of complexity to the task. The field of sarcasm detection has made great strides with the use of supervised classification models that can accurately distinguish between sarcastic and non-sarcastic sentences through labelled data. This approach shows promise for identifying explicitly incongruous statements, and there is an increasing need for techniques that can successfully detect sarcasm in sentences with implicit sentiment incongruity. This study employs a deep learning model to tackle the challenge of distinguishing implicit sarcasm. The model being used is a recurrent neural network (RNN), which has the ability to retain numerical representations of previously processed information. The proposed model is designed to effectively identify sarcasm by capturing context-based incongruity within the presented text, which is a crucial factor. The proposed model, Proposed LSTM (P-LSTM), effectively captures the dependencies among words in both suffixes and prefixes by scanning the given sentences bi-directionally. The classification is organised into two levels: emotional and semantic. The suggested approach offers a way to accurately evaluate and categories tweets based on their sarcastic state. Excitingly, the proposed method is being evaluated through experiments on two automatically annotated datasets and two manually annotated datasets to determine its effectiveness. The suggested model is compared to numerous state-of-the-art methods, highlighting its efficacy. The findings demonstrate that the suggested model is highly effective for sarcasm detection compared to other methods currently in use. The suggested model has great potential in recognising implicit sarcasm by utilising deep learning and context-based incongruity analysis, which could lead to significant improvements over existing methods.

* *Corresponding author.*
E-mail address: mkmuddada@lincoln.edu.my

https://doi.org/10.37934/araset.31.2.352362

1. Introduction

In social media, the client-side users create text messages that are very creative and contain a high level of embedded emotions like sarcasm and other similar expressions. The most prominent social networking platforms, such as Twitter, Facebook, and Instagram, as well as others, all make use of a feature known as "Hashtag (#)," which allows users to create a tag of their own and give it a moniker of their choosing. There is no requirement that it be expressed via formal language or representation. In the process of data mining and analysis, the identification of caustic remarks is of the utmost importance. The insensitivity of assigning attention to caustic words might result in the opposite of what is intended [19]. The identification of sarcasm within the comments is also crucial for reasons relating to security [24]. Finding sarcasm inside a text is a challenging process that demands a high degree of intelligence since, unlike data mining from text messages, it is difficult to recognize sarcasm in the text. Data mining from text messages may be done automatically. Sometimes the material will contain indirect caustic comments, and the only way to recognize them is after a discussion has taken place.

As per the researchers of linguistics, sarcasm is an incongruity conveyed between the statement and the context that it refers to within a single piece of text [19, 25, 26]. Contrast from sentiment polarity is one among the common form of sarcasm in social media Twitter. For instance, “Am astonished to see these dirty environment” in this tweet, astonished refers to positive polarity and dirty environment indicates negative polarity. Here in this tweet the sarcasm lies in dirt environment and its strongness of sarcasm lies on the positive polarity astonishing. There are basically two forms of incongruity 1. Explicit congruity: for instance, “I love annoying” here love is positive and annoying is negative which comes under sentiment polarity, and this indicates the sarcasm directly. Whereas implicit sarcasm lies between the words that are highly undistinguishable as words of sarcasm.

The task in detection of sarcasm is to identify and classify a piece of text whether it belongs to sarcastic or non-sarcastic. Even though it is a single piece of text that are supposed to be classified, the task highly depends on the words that are supported for it. General machine learning models can classify the explicit sarcasm in an effective way. However, this coupling and dependency of words and its meaning among each other in the sentence are hard to classify. If the dependency were on sequential form, then a general Recurrent Neural Network would easily resolve it.

However, RNN is able to categorize the sentences depending on the order in which they appear sequentially. It does not proceed backward and forward in order to determine the second level meaning that is included inside the phrase. Scanning the assertions back and forth like this provides an effective method that is discussed in this study for addressing the detection of sarcastic text that is included in the phrases. The suggested model, which will be called P-LSTM, is made up of two levels of classification: the first level focuses on the emotional aspect, while the second level focuses on the semantic aspect.

The detailed studies on the recent models are discussed in section 2. The basic explanations, which are highly needed for the further understanding of the concepts in this paper, can be, and found in section 3. In Section 3, we cover the fundamentals of RNN and the specifics of Bi-directional LSTM. Section 4 provides an example of using Bi-directional LSTM to detect sarcasm in text. Extensive information on the models used to demonstrate the importance of the proposed model is provided, and Section 5 explains the experimental setup, experimental analysis, and datasets selected for evaluating the performance of the proposed model. Section 6concludes with a discussion on how the current work might be improved in the future.

2. Related Works

In recent years the rule-based approach to classify the sentiments on the statements in which it highly dependent on the linguistics and the models that depends on such approaches includes [25, 27, 28]. In Twitter social media platform, the usage of hashtags can reverse the appropriate sentiment in the tweets present over there [28]. A direct hashtag called #sarcasm is used in most of the tweets to directly indicate that this text belongs to sarcasm. These hashtags are not present in some of the other aspects where the task become highly difficult and the false positive in the comments of #sarcsam should also be identified and removed.

In the current decade the use of deep learning approaches is high in demand where without the intervention of the user the model can distinguish between the sarcastic and non-sarcastic statements. Few models use the similarity score to distinguish the same. In the year 2016 [29], a machine learning approach is employed to find the sarcastic detection on a domain level classification. CNN based classification models are also present in the sarcasm detection domain.

There are a substantial variety of methods that may be employed in research, such as the direct CNN approach, the direct LSTM approach, the LSTM fused with the CNN approach, the Bi-LSTM method, and many more. In the course of this study effort, comparisons were conducted between the CNN-based strategy, the LSTM-based technique, the CNN-LSTM combined approach, and the Bi-LSTM approach. There are just a few additional works that are relevant to this topic in [20-23].

3. Bi-directional RNN

In this section, the working model of Bi-directional RNN and the working model of Bi-Directional Long Short-Term memory based neural network is explained for the clear understanding for further readings. The Bi-directional RNN is the base model for the working of Bidirectional LSTM.

3.1 Bi-directional RNN

In the conventional neural network models, the input of every layer is independent to each other and the same is applicable to the output and hidden layers. However, in sentiment analysis, for identifying the actual emotion in the sentences, the cohesion between the words in the sentences is also essential. Based on this model even the missing out emotions can also be discovered provided with appropriate training to the model. The remembrance of correlation between the words in the sentences it can transfer the actual emotions in the sentences from input layer to the hidden layer for further processing.

The working of RNN is as follows: RNN loops the networks among them through which it can retain the core content of information. Since the loop is nested in structure it has the tendency to accept the input as a sequence.

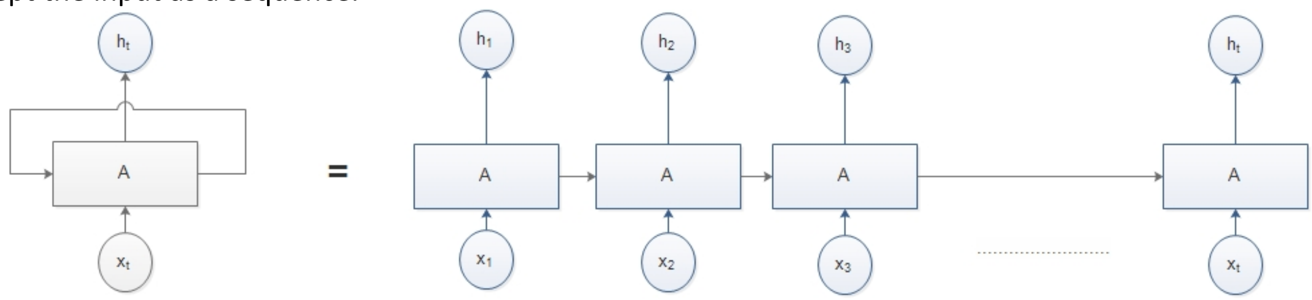

Fig. 1. RNN working Model

RNN transfers the independent working model of CNN into dependent actions. For instance, the activation functions which are independent in CNN are dependent to each other in RNN. The weights and the bias values in all the layers will be the same from one another. In this way the computational complexity will be reduced measurably. In this model, the output of the first layer will be the input for the next hidden layer. In this model of computation, all these three layers (input, hidden and output) will be considered as a single layer.

In figure 1, h represents the hidden layer, x represents the input, $x_1, x_2 \dots x_n$ represents the input words in the sequence where each x_i represents one word each.

The calculation of information extracted from the input sequence i can be mathematically represented as,

$$h_i = f(h_{i-1}, x_i) \tag{1}$$

where i is the current iteration. And when applying the activation function ($\tanh$) the model can be represented mathematically as

$$h_i = \tanh\left(w_{ii}h_{i-1} + w_{ix}x_i\right) \tag{2}$$

where ii represents the recurrent neuron and ix represents the current neuron. The final output can be calculated as

$$y_i = w_{hy} \times h_i \tag{3}$$

whereas y denotes the output of the neuron and $w_{hy}$is the weight of the output layer neuron.

3.2 Bi-Directional LSTM

The bi-directional LSTM is the fusion of two independent RNN together. With this model it is effective to appropriately examine the sentence both forward and backward at same time. In Bi-directional LSTM, the input will be taken in two different RNN in which one runs forward form and the other in the backward form.

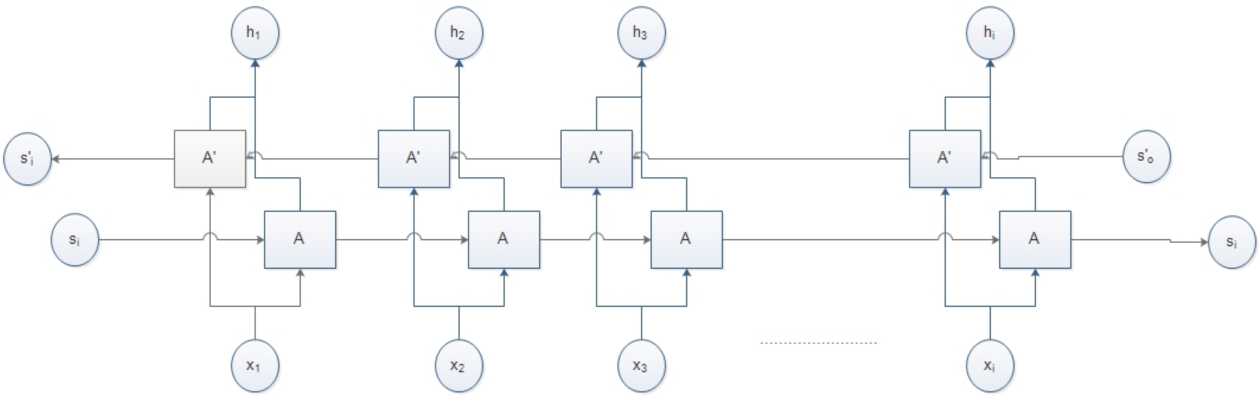

Fig. 2. Bi-directional LSTM

In Bi-directional LSTM the calculation of activation functions do not happen in s single candidate value solution and the calculation of such values can be mathematically formulated as

$$C^i = \tanh\left(w_c\left[a^{i-1}, x^i\right] + b_c\right) \tag{4}$$

The control over LSTM network can be carried out in three different forms:

1. To update the gates in the network

$$\delta_u = \sigma\left(w_u\left[a^{i-1}, x^i\right] + b_u\right) \tag{5}$$

2. To forget the sequence of words in the network

$$\delta_f = \sigma\left(w_f\left[a^{i-1}, x^i\right] + b_f\right) \tag{6}$$

3. The output of the network can be computed as

$$\delta_o = \sigma(w_o[a^{i-1}, x^i] + b_o) \tag{7}$$

And the final output can be represented as

$$output = \begin{cases} c^i = \delta_u \times c^i + \delta_f \times c^i \\ a^t = \delta_0 \times c^i \end{cases} \tag{8}$$

For example: the findings from the surface pressure measuring experiment are discussed in this section. In the next sub section, discusses the impacts of the angle of attack, the Reynolds number, and the leading edge bluntness.

4. Bi Directional-RNN for detecting Sarcasm

The knowledge may be effectively transformed from one situation to another by using learning models, which is quite successful due to the fact that the memory of the network can be obviously utilised in multiple various aspects. Identifying sarcasm within the supplied material as a statement and its relation to the context of it is a knowledge-based learning mechanism that was utilised in this study effort. The sources that were utilised for this mechanism were Twitter and other online social media platforms.

There is a large selection of resources for determining sentiments that can be found on the internet. Some examples of these include sentiment word corpora [1, 2] and sentiment tweets corpora [3, 4]. In order to strengthen the suggested models, a few more models were also brought to the attention of the research team. These models transmit the knowledge of feelings gained from other resources obtained from the above-mentioned corpora [1, 2, 5].

The exiting challenge in this model is to identify the coupling form of two different sentiment analysis technique to find the sarcasm present in the statements given. The two identified models are sentiment word corpora and sentiment tweets corpora.

One is to find the sentiment hard attention from every word in the sentence. And the other is to find the semantic present in it. Both can be mathematically represented as follows.

The representation of sentiment analyzed from the tweets can be represented as $y = [y_1^*, y_2^*, y_3^* \dots, y_i^*]$ and this is generated after applying the soft-max (tanh) function in it from S.

$$y_i^* = \tanh(|S|) \tag{9}$$

$$\mathbb{R} = H(y \oplus W \times y^*) \tag{10}$$

5. Experimental Analysis

The experimental design, datasets used in the assessment, and a comparison of the proposed model's performance to that of current models are all detailed here.

Experimental Setup

We test our proposed model by using the following table 1 of automatically annotated and two manually annotated datasets.

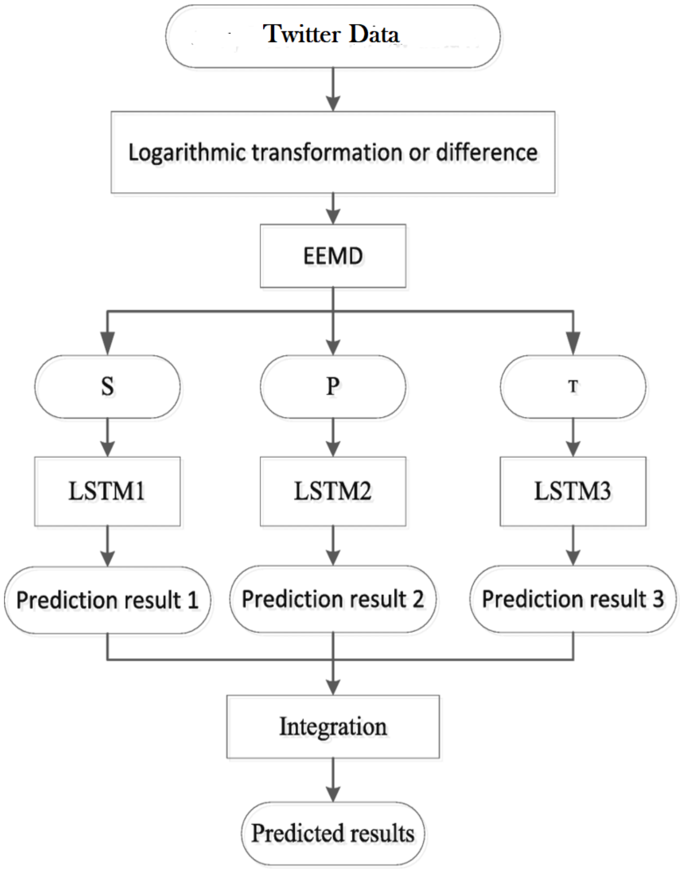

Fig. 3. Working flow of LSTM on sarcasm classification

Table 1
Datasets

Si. No.	Datasets	Reference	# Sarcastic / # Non-Sarcastic Statements
1	Automatically Annotated Dataset	[6]	10000 / 10000
2		[7]	10000 / 10000
3		[8]	18889 / 48890
4		[22]	2826 / 1792
5	Manually annotated dataset 1	[9]	474 / 1608
6		[10]	532 / 1397
7		[11]	2222 / 2396

The compared algorithms for proving the performance of the proposed model includes Bi-LSTM networks [12, 13, 14], CNN based LSTM networks [15], LSTM [16], CNN [17] feature based sarcasm detection models [9, 18, 19].

5.1 Experimental results and discussion

The experimental analysis was done in three stages. 1. On automatically annotated datasets which uses hashtags to find the instances.

5.2 *Automatically Annotated Datasets*

P-LSTM (proposed LSTM) findings on automatically annotated datasets are shown in Table 2. The outcomes are evaluated next to the most up-to-date algorithms. Hashtags like #sarcasm, #education (edu), #humour (hum), #politics (pol), and #news are used to automatically extract occurrences from the dataset.

Table 2
Results of Proposed P-LSTM vs other Algorithms on Automatically Annotated Datasets

Models	[6]			[7]			
	edu	hum	pol	edu	hum	poi	news
[14]	94.73	96.26	97.04	94.84	95.58	99.34	96.96
[15]	93.77	94.46	95.06	94.88	96.51	99.21	97.83
[16]	93.85	91.05	90.55	95.58	95.78	99.11	97.66
[17]	94.28	94.37	95.59	95.05	96.46	99.24	97.21
P-LSTM	94.83	95.83	96.69	95.09	96.28	98.76	97.06
[20]	91.12	91.12	93.12	91.12	93.12	95.12	97.12

Table 2 shows that the suggested model achieves better results than the state-of-the-art algorithms do for the vast majority of the examined hashtags. Numerical results show that the proposed P-LSTM is superior than [14], [15], and [19] and is on par with [16] and [17] across the board for hashtags. The data presented in Table 2 are given a graphical representation in Figure 4, which facilitates easier comparison.

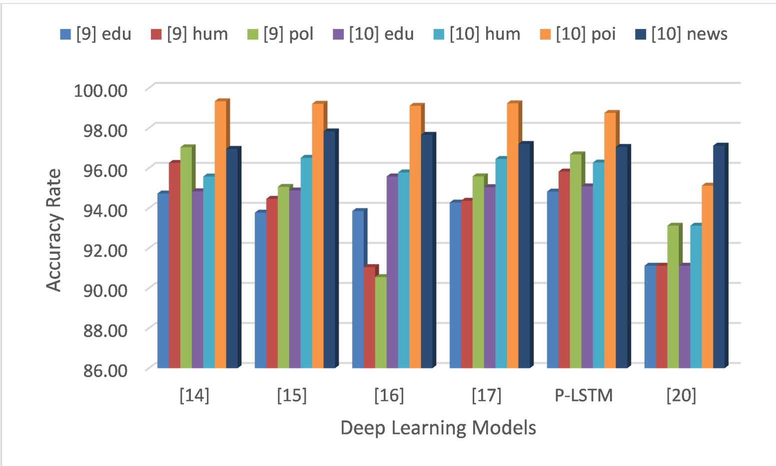

Fig. 4. Results of Proposed P-LSTM vs other Algorithms on Automatically Annotated Datasets

5.2.1 Manually annotated datasets

The outcomes of applying P-LSTM (proposed LSTM) on manually annotated datasets are presented in Table 3. The findings are analysed in light of the most current algorithmic developments. Instances from the dataset are extracted by hand, and the process is described in the articles [9, 10, 11, 22].

Table 3
Results of Proposed P-LSTM vs other Algorithms on Manually Annotated Datasets

	[9]	[10]	[11]	[22]
[14]	75.39	60.13	65.97	69.53
[15]	71.29	59.95	61.89	68.07
[16]	72.70	57.69	64.19	68.54
[17]	76.01	58.97	63.29	70.70
P-LSTM	78.42	67.37	69.57	73.20

It is clear from looking at Table 3 that the suggested model performs significantly better than the methods that are already in use. Based on the numerical results, it is possible to claim that the proposed P-LSTM outperforms [14], [16], and [17] and performs similarly well as [15] in all manually annotated datasets. In addition, the suggested P-LSTM performs better than [15]. The data presented in Table 3 are depicted in a graphical format in Figure 5, which facilitates easier comparison.

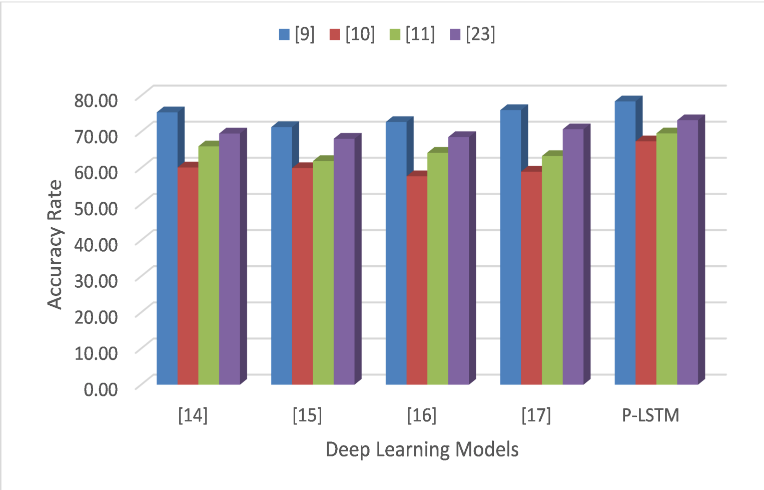

Fig. 5. Results of Proposed P-LSTM vs other Algorithms on Manually Annotated Datasets

5.2.2 Unbalanced Datasets

The results of the P-LSTM (proposed LSTM) algorithm applied to imbalanced datasets are presented in Table 4. The findings are analysed in light of the most current algorithmic developments. The instances of the dataset are retrieved by hand, and the comparisons are carried out using sets of data that are not evenly distributed. For instance, roughly 30000 non-sarcastic comments are utilised for training the networks in addition to 10000 sarcastic remarks [6, 7, 8]. This is done so that the networks may learn from both types of comments.

It is clear from looking at Table 4 that the suggested model performs significantly better than the methods that are already in use. Based on the numerical results, it is possible to claim that the proposed P-LSTM performs better than [14], [19], and [16] in all imbalanced datasets, while performing similarly well as [15] and [17]. The data presented in Table 4 are depicted in a graphical format in Figure 6, which allows for easier comparison.

Table 4
Results of Proposed P-LSTM vs other Algorithms on unbalanced Datasets

	[6]	[7]	[8]
[14]	92.50	95.81	98.51
[15]	92.23	95.43	98.99
[16]	92.16	94.85	98.25
[17]	92.97	95.39	99.32
P-LSTM	92.93	96.30	99.80
[20]	92.89	95.47	99.20

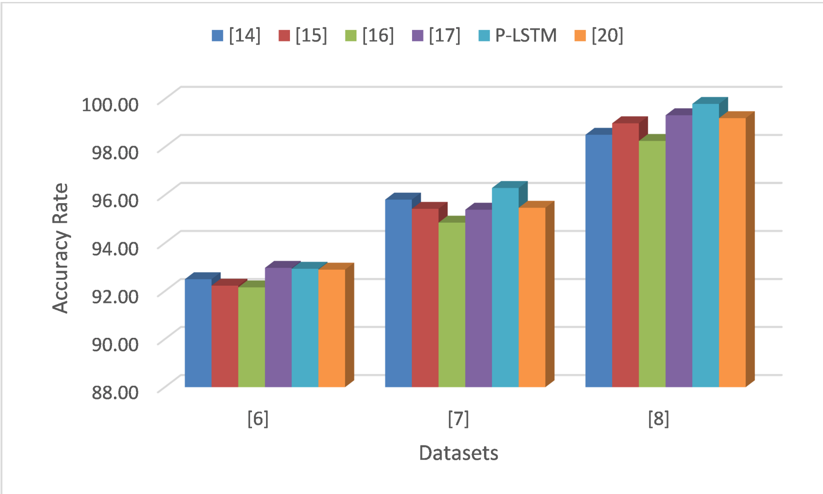

Fig. 6. Results of Proposed P-LSTM vs other Algorithms on unbalanced Datasets

6. Conclusion

A innovative solution to the problem of identifying sarcasm in tweets is presented here as part of a research project on the subject. The model is constructed from an approach to deep learning known as RNN, in which the records of the previously scanned material are retained in the form of numbers. The suggested model's primary focus should be on the inconsistency of the supplied texts with regard to their respective contexts. The suggested P-LSTM works by scanning the provided phrases to and from inside the model. This enables the information and the dependency among the words in suffix as well as prefixes to be uncovered. After that, the suggested model is assessed based on the classification-trained model, and the tweets are categorized based on the accuracy ratio. In order to demonstrate that the suggested model is effective, it is examined and contrasted with the most recent algorithm that represents the state of the art. The findings demonstrated that the suggested model is superior to other current approaches in terms of its overall efficacy.

Acknowledgement
The authors want to extend their gratitude to everybody and everyone who assisted them while they were working on this project. The authors of this study did not get any special fund for their

endeavour. Regarding the current research, the authors have stated that they do not have any potential conflicts of interest to report.

References

[1] Baccianella, Stefano, Andrea Esuli, and Fabrizio Sebastiani. "Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining." In *Lrec*, vol. 10, no. 2010, pp. 2200-2204. 2010.

[2] Hutto, Clayton, and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text." In *Proceedings of the international AAAI conference on web and social media*, vol. 8, no. 1, pp. 216-225. 2014. https://doi.org/10.1609/icwsm.v8i1.14550

[3] Ghosh, Aniruddha, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, John Barnden, and Antonio Reyes. "Semeval-2015 task 11: Sentiment analysis of figurative language in twitter." In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pp. 470-478. 2015. https://doi.org/10.18653/v1/S15-2080

[4] Rosenthal, Sara, Noura Farra, and Preslav Nakov. "SemEval-2017 task 4: Sentiment analysis in Twitter." *arXiv preprint arXiv:1912.00741* (2019). https://doi.org/10.18653/v1/S17-2088

[5] Wilson, Theresa, Janyce Wiebe, and Paul Hoffmann. "Recognizing contextual polarity in phrase-level sentiment analysis." In *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, pp. 347-354. 2005. https://doi.org/10.3115/1220575.1220619

[6] Reyes, Antonio, Paolo Rosso, and Tony Veale. "A multidimensional approach for detecting irony in twitter." *Language resources and evaluation* 47 (2013): 239-268. https://doi.org/10.1007/s10579-012-9196-x

[7] Barbieri, Francesco, Horacio Saggion, and Francesco Ronzano. "Modelling sarcasm in twitter, a novel approach." In *proceedings of the 5th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pp. 50-58. 2014. https://doi.org/10.3115/v1/W14-2609

[8] Ptáček, Tomáš, Ivan Habernal, and Jun Hong. "Sarcasm detection on czech and english twitter." In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers*, pp. 213-223. 2014.

[9] Riloff, Ellen, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. "Sarcasm as contrast between a positive sentiment and negative situation." In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 704-714. 2013.

[10] Mohammad, Saif M., Xiaodan Zhu, Svetlana Kiritchenko, and Joel Martin. "Sentiment, emotion, purpose, and style in electoral tweets." *Information Processing & Management* 51, no. 4 (2015): 480-499. https://doi.org/10.1016/j.ipm.2014.09.003

[11] Van Hee, Cynthia, Els Lefever, and Véronique Hoste. "Semeval-2018 task 3: Irony detection in english tweets." In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 39-50. 2018. https://doi.org/10.18653/v1/S18-1005

[12] Felbo, Bjarke, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. "Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm." *arXiv preprint arXiv:1708.00524* (2017). https://doi.org/10.18653/v1/D17-1169

[13] Ghosh, Debanjan, Alexander Richard Fabbri, and Smaranda Muresan. "The role of conversation context for sarcasm detection in online interactions." *arXiv preprint arXiv:1707.06226* (2017). https://doi.org/10.18653/v1/W17-5523

[14] Zhou, Peng, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. "Attention-based bidirectional long short-term memory networks for relation classification." In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)*, pp. 207-212. 2016. https://doi.org/10.18653/v1/P16-2034

[15] Ghosh, Aniruddha, and Tony Veale. "Fracking sarcasm using neural network." In *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pp. 161-169. 2016. https://doi.org/10.18653/v1/W16-0425

[16] Huang, Yu-Hsiang, Hen-Hsen Huang, and Hsin-Hsi Chen. "Irony detection with attentive recurrent neural networks." In *Advances in Information Retrieval: 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, April 8-13, 2017, Proceedings 39*, pp. 534-540. Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-56608-5_45

[17] Kim, Yoon. "Convolutional neural networks for sentence classification." *arXiv preprint arXiv:1408.5882* (2014). https://doi.org/10.3115/v1/D14-1181

[18] Joshi, Aditya, Vinita Sharma, and Pushpak Bhattacharyya. "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)*, pp. 757-762. 2015. https://doi.org/10.3115/v1/P15-2124

[19] Thirugnansambandam, Kalaipriyan, Debnath Bhattacharyya, JaroslavFrnda, Dinesh Kumar Anguraj, and Jan Nedoma. "Augmented Node Placement Model in -WSN Through Multiobjective Approach." *Computers, Materials & Continua* 69, no. 3 (2021), 3629-3644. https://doi.org/10.32604/cmc.2021.018939

[20] Farías, Delia Irazú Hernańdez, Viviana Patti, and Paolo Rosso. "Irony detection in twitter: The role of affective content." *ACM Transactions on Internet Technology (TOIT)* 16, no. 3 (2016): 1-24. https://doi.org/10.1145/2930663

[21] Thirugnanasambandam, Kalaipriyan, Raju Anitha, Vamsidhar Enireddy, R. S. Raghav, Dinesh Kumar Anguraj, and Amirthasaravanan Arivunambi. "Pattern mining technique derived ant colony optimization for document information retrieval." *Journal of Ambient Intelligence and Humanized Computing* (2021): 1-13. https://doi.org/10.1007/s12652-020-02760-y

[22] Thirugnanasambandam, Kalaipriyan, S. V. Sudha, D. Saravanan, Renjith V. Ravi, Dinesh Kumar Anguraj, and R. S. Raghav. "Reinforced Cuckoo Search based fugitive landfill methane emission estimation." *Environmental Technology & Innovation* 21 (2021): 101207. https://doi.org/10.1016/j.eti.2020.101207

[23] Anguraj, Dinesh Kumar, Kalaipriyan Thirugnanasambandam, Raghav RS, and Sudha SV. "Enriched cluster head selection using augmented bifold cuckoo search algorithm for edge-based internet of medical things." *International Journal of Communication Systems* 34, no. 9 (2021): e4817. https://doi.org/10.1002/dac.4817

[24] González-Ibánez, Roberto, Smaranda Muresan, and Nina Wacholder. "Identifying sarcasm in twitter: a closer look." In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pp. 581-586. 2011.

[25] Gerrig, Richard J., and Yevgeniya Goldvarg. "Additive effects in the perception of sarcasm: Situational disparity and echoic mention." *Metaphor and Symbol* 15, no. 4 (2000): 197-208. https://doi.org/10.1207/S15327868MS1504_1

[26] Ivanko, Stacey L., and Penny M. Pexman. "Context incongruity and irony processing." *Discourse processes* 35, no. 3 (2003): 241-279. https://doi.org/10.1207/S15326950DP3503_2

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

[28] Sulis, Emilio, Delia Irazú Hernández Farías, Paolo Rosso, Viviana Patti, and Giancarlo Ruffo. "Figurative messages and affect in Twitter: Differences between# irony,# sarcasm and# not." *Knowledge-Based Systems* 108 (2016): 132-143. https://doi.org/10.1016/j.knosys.2016.05.035

[29] Poria, Soujanya, Erik Cambria, Devamanyu Hazarika, and Prateek Vij. "A deeper look into sarcastic tweets using deep convolutional neural networks." *arXiv preprint arXiv:1610.08815* (2016).