iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** III **Month of publication:** March 2023

DOI: https://doi.org/10.22214/ijraset.2023.49378

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue III Mar 2023- Available at www.ijraset.com

Sarcasm Detection for Hindi English Code Mixed Twitter Data

Asst. Prof. Koti Tejasvi[1], Borra Richa Reddy[2], Sheri Sukhjeevan Reddy[3], Sirisilla Rishikesh[4]
[1, 2, 3, 4]*Computer science and engineering Vardhaman College of Engineering Telangana, India*

Abstract: ***Twitter, Instagram have grown to be the two most popular social media sites for users to voice their opinions on a variety of subjects. The generation of such huge amounts of user data has made NLP activities like sentiment analysis and opinion mining far more important. Sarcastic statements on social media, also referred to as memes, have recently become very popular. Sarcasm challenges many NLP tasks because it flips the meaning and polarity of what the language implies. A lot of resources were developed for the English language, but this does not hold true for Hinglish In this paper we include a tweet corpus for training unique word embeddings as well as a Hinglish dataset labelled for sarcasm detection. Although there have been various attempts to categorise a text's sentiment, there aren't many models that can do the same when given non-English data that contains sarcasm or irony . This study compares numerous sarcasm detection methods for Hinglish data in order to determine which approach performs the best on datasets of various sizes and types. We have presented a technique that will enhance the outcomes of sarcasm recognition for Hindi- English code-mixed tweets by examining and researching the prior work.***

I. INTRODUCTION

Global social network users are predicted to reach 3 billion in 2024 as they continue to increase. The number of people using social, interactive computer-mediated platforms like Twitter, Tumblr, Google+, Facebook, Instagram, Snapchat,and others that allow users to create, upload, and share various types of multimedia text is growing. Social media acts as a route for communication as well as a tool for social listening, awareness, activism, and feedback to encourage cooperation among stakeholders. Monitoring social media sentiment, sometimes known as "the online mood," is an important aspect of social media listening [3]. To gauge and report on the tone or sentiment of your social mention, sentiment analysis is a crucial part of the social listening tool. Natural language processing (NLP) is used to examine social media conversations and ascertain more detailed context for a mention. One such sentiment is sarcasm. Sarcasm is definedas a cutting, often ironic remark intended to express contempt or ridicule. Correctly classifying a text as "sarcastic" or "non-sarcastic" is the challenge of sarcasm detection. It isa difficult task because text lacks tone and facial emotions, especially when it is code mixed data. Despite the fact that English is the language that is used the most frequently on these websites, the majority of users are not native English speakers. As a result, the majority of people text in their own language. This opens up the possibility of interacting withdata produced in multiple languages by social media websites [10]. Numerous statistics indicate that about 26 percent of Indians are bilingual. The phenomenon of code switching and code mixing results from this. Code mixing takes place when speakers use two or more languages below clause level in a single social context.

Tweet: gasoline par chalne wali ek nayi suv banayi hai fordne/ ford ne ek nayi suv banayi jo gasoline par chalti haiTranslation: ford develops new suv that runs purely ongasoline Sarcasm: YES

This paper's contribution is to offer a collection of tweetsthat are combined in English and Hindi and contain bothsarcastic and non-sarcastic tweets. The proposed models take self-trained bilingual word embeddings generated by Hindi- English code mixed data as input [8]. In order to identify sarcasm, Deep learning is being used extensively in the domain of natural language processing and has given satisfactory results.

Our work has made several contributions, including:

1) In this study, we tested deep learning methods to identify sarcasm in a dataset.
2) In the field of natural language processing, deep learningis widely employed and has shown positive results. Different deep learning models, including Bi-directional LSTM, and Bi-directional GRU are proposed in our work.
3) Self-trained bilingual word embeddings produced from mixed Hindi-English code data are the input for the suggeste models.
4) Our research offers an alternative methodology to previous research utilising SVMs and random forests, two well-known traditional machine learning methods.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue III Mar 2023- Available at www.ijraset.com

II. RELATED WORK

A Support Vector Machine (SVM)-based sarcasm detector for Hindi texts was proposed by Desai and Dave . Hindi tweets were utilised as the dataset for the SVM classifier's training [2] and testing. They translated English tweets into Hindi because they didn't have access to annotated datasetsfor training and testing. In order to identify sarcasm in Englishtext, they concentrated on a comparable collection of elementslike emoticons and punctuation. These techniques are not directly used for the naturally occurring sarcastic tweets in Hindi as displayed in Figure 1.

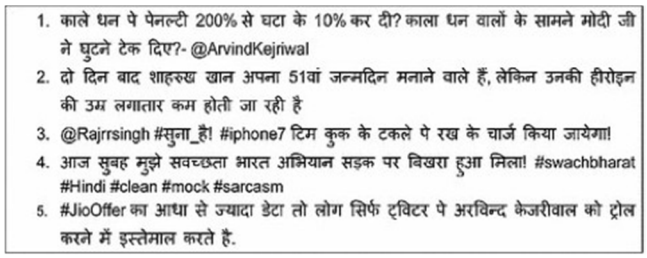

1. काले धन पे पेनल्टी 200% से घटा के 10% कर दी? काला धन वालों के सामने मोदी जी ने घुटने टेक दिए?- @ArvindKejriwal
2. दो दिन बाद शाहरुख खान अपना 51वां जन्मदिन मनाने वाले हैं, लेकिन उनकी हीरोइन की उम्र लगातार कम होती जा रही है
3. @Rajrrsingh #सुना_है! #iphone7 टिम कुक के टकले पे रख के चार्ज किया जायेगा!
4. आज सुबह मुझे सवच्छता भारत अभियान सड़क पर बिखरा हुआ मिला! #swachbharat #Hindi #clean #mock #sarcasm
5. #JioOffer का आधा से ज्यादा डेटा तो लोग सिर्फ ट्विटर पे अरविन्द केजरीवाल को ट्रोल करने में इस्तेमाल करते है.

Fig.1. Sample tweets

III. PROPOSED METHODOLOGY

A. Dataset Creation

In the absence of sufficient dataset for training and testing, detection of sarcastic sentiment is a challenging task in Hindi. This article proposed a context-based pattern for sarcasm detection in Hindi tweets. The proposed approach attains an accuracy of 82.55 percent. The proposed approach outperformsthe state-of-the-arts techniques for sarcasm detection in Hindi tweets. A readily buildup dataset of 30000 tweets. [8]Out of these, the 28619 tweets are extracted manually to create a balanced dataset of 12000 sarcastic and 5000 non-sarcastic tweets that are considered as the dataset for this research. Oncethis is done now we have translated all the Hindi tweets from the dataset into English in order to create word embeddings.

B. Data Preprocessing

Social media data is extremely noisy, necessitating exten- sive preparation. We eliminated any mentions (@) and the 'hashtags' symbols from the data while creating the dataset. In order to prevent our deep learning models from being biassed towards particular terms during learning, we also deleted search tags (such as cricket and sarcasm) and stop words (words that have no meaning but are utilised as partsof English grammar). Also, punctuation marks and URLs weretaken out.

C. Creation of Word Embeddings

Being a text classification problem, it is essential for the words of the dataset to be first converted to vector representations. Word embedding is learned from unannotated plain text, useful in determining the context in which a given word is used. They provide a dense vector representation of syntactic or semantic aspects of a word.

We experimented with two different kinds of word embeddingson the dataset, which include,

1) *Word2Vec:* The word2vec algorithm uses a neural network model to learn word associations from a large corpus of text. Words from the corpus are turned into vectors in this embedding, and words with similar contexts are positioned close to one another in the vector space. Once trained, such a model can detect synonymous words or suggest additional words for a partial sentence.
2) *GloVe (Global Vectors):* The GloVe is a model that is applied to the distribution of words. The distance betweenthe terms in this unsupervised learning algorithm's word map indicates how semantically related the words are to one another. These algorithms train a corpus made up of aggregated global word-word co-occurrence statistics. The trained corpus typically corresponds to the subspace of the words that are of interest to us.

D. Training Models

We have not limited our comparison only to deep learning models; we have also simultaneously compared the performance of several machine learning models (Multinomial Naive Bayes, Stochastic Gradient Descent, KNeighbours, Logistic Regression, Decision Tree, Random Forest, Support Vector Classification) along with deep learning models(Bidirectional LSTM, Bidirectional GRU) in order to find the best possible algorithm that is well suitedfor our dataset.

The models which we have used includes,

1) Machine Learning Models

a) Stochastic Gradient Descent Classifier

To determine the parameters or coefficients of functions that minimise a cost function, one can use the straightforward yet effective optimization process known as stochastic gradient descent (SGD). To put it another way, it is employed in the discriminative learning of linear classifiers under convex loss functions, including SVM and logistic regression. Because the update to the coefficients is done for each training instance rather than at the end of examples, it has been successfully used to large-scale datasets.

b) Multinomial Naive Bayes Classifier

One of the Naive Bayes algorithm modifications used in machine learning is known as multinomial Naive Bayes, and it is excellent for usage on multinomial distributed datasets. This algorithm can be used to predict the label of the textwhen there are several classes to categorise the text into. It does this by calculating the probability of each label forthe input text and then producing the label with the highest probability as the output. A series of probabilistic algorithms known as the Naive Bayes Classifier Algorithm is based on using the Bayes Theorem with the ”naive” assumption that each pair of features is conditionally independent.

c) K Neighbors Classifier

The k-nearest neighbours algorithm, sometimes referred to as KNN or k-NN, is a supervised learning classifier that employs proximity to produce classifications or predictions about the grouping of a single data point. Although it can be applied to classification or regression issues, it is commonly employed as a classification algorithm because it relies on the idea that comparable points can be discovered close to one another. The K-NN algorithm makes the assumption that the new case and the existing cases are comparable, and it places the new instance in the category that is most like the existing categories. The KNN method simply saves the information during the training phase, and when it receives new data, it categorises it into a category that is quite similar to the new data.

d) Decision Tree Classifier

A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. By using a greedy search to find the ideal split points inside a tree, decision tree learning uses a divide and conquer technique. When most or all of the records have been classified under distinct class labels, this splitting procedure is then repeated in a top-down, recursive fashion. The complexity of the decision tree plays a significant role in determining whether or not all data points are categorised as homogenous sets. Pure leaf nodes, or data points belongingto a single class, are easier to obtain in smaller trees. It gets harder to preserve this purity as a tree gets bigger, which typically leads to too little data falling under a particular subtree.

e) Logistic Regression Classifier

A classification process called logistic regression is used to determine the likelihood that an event will succeed or fail. When the dependent variable is binary (True/False, Yes/No, 0/1, etc.), it is used. By analysing the relationship from agiven collection of labelled data, it helps classifying data into distinct classes. From the provided dataset, it learns a linear relationship before introducing a non-linearity in the form of the Sigmoid function.

f) Support Vector Classification

One of the supervised learning methods, Support Vector Machine is employed in both organisation and regression scenarios. It was initially employed to solve categorisation issues in machine learning.

The goal of the Support Vector Machine method is to create decision boundaries that divide n-dimensional space into categories so that fresh data centres can be readily placed in the ideal class in the future. Theideal decision boundary should be regarded as hyper plane. This support vector machine approach can also be used for face detection, image classification, and text classification,among other things.

g) Random Forest Classifier

A randomly chosen portion of the training data is used by the Random forest classifier to generate a collection of decision trees.

It simply consists of a collection of decision trees drawn from a randomly chosen subset of the training set,which are then used to decide the final prediction. One of the supervised learning algorithms used in machine learning for grouping and regression issues is known as random forest. However, the organisation scenarios are the key use. We all know that a forest is full of trees, and that when there aremore trees, the forest is said to be fully healthy. Similar tothis, the random forest approach in machine learning creates decision trees from the examples provided before predicting the result that will be most appropriate. By combining the outcomes of several decision trees, it typically resolves the issue of overfitting. More data items are processed by the random forest machine learning algorithm than by a single decision tree.

2) Deep Learning Models

a) Bi-lstm

Bidirectional long-short term memory (bi-lstm) is the technique of allowing any neural network to store sequence information in both directions, whether they are backwards (from future to past) or forward (past to future). A bi-lstm differs from a standard LSTM in that our input flows in two directions when it is bidirectional.

We can direct input toflow in either a forward or a backward direction using the standard LSTM. To maintain both the future and the past knowledge, bi-directional input can be made to flow both ways. When used for text categorisation tasks, bi-directional LSTMs have been successful in capturing the context . Aword's context is influenced not only by the words that come before it, but also by the words that come after it [9]. Tomodel this, memory cells are needed in both the forward and backward directions, with cells in the forward direction keeping track of words' histories. Two LSTM layers are added to the input embedding layer to do this and capture the composition semantics of data. Following concatenation, the output features from the two layers are flattened and supplied to two dense fully connected layers. Like in all other models,a single neuron is responsible for classification. The BiLSTM computes the forward hidden sequence by traversing the forward layer from time t=1 to time t=T, and the backward hidden sequence by traversing the backward layer from time t=T to time 1, and updates the output.

b) BiGRU

We must first comprehend GRU in order to comprehend Bidirectional GRU . GRU is a less complex version of the LSTM network. In contrast to the three gates of the LSTM (input gate, ignoring gate, and output gate), GRU only hastwo gates(update gate and reset gate). Update gates serve identical purposes to input and forgetting gates in LSTM. It chooses which information to keep and which to refresh and add new information. The reset gate is used to determine whichaspect of the prior knowledge is irrelevant to the calculationof the present time. GRU performs calculations more quickly than LSTM because it has fewer gates than the latter. [13] Bidirectional GRU's model layout is comparable to that of theGRU model. Both a positive and a reverse time sequence exist. The final output results are the results that correlate to the finalstate of the positive time series and the reverse time series. Themodel can simultaneously use knowledge from the past and thefuture. We employ the bidirectional GRU model in this work. The forward status and backward status subnetworks stand for forward transmission and reverse transmission, respectively. A Bidirectional GRU, or BiGRU, is a sequence processing model that consists of two GRUs. one taking the input in a forward direction, and the other in a backwards direction. Itis a bidirectional recurrent neural network with only the input and forget gates.

IV. MAKING PREDICTIONS

The best performing algorithm in terms of accuracy and efficiency will be determined after training utilising the aforementioned methods, and it will then be used to create our prediction system. Hindi, English, or mixed Hindi and English text is to be provided as input in order to create predictions. The given input is then translated into English using the translation tool, and then the input text is preprocessed. Following preprocessing, the input is fed into the chosen model, and a numerical forecast is provided. The output obtained is now given a numerical threshold based on which it is determined whether or not it is sarcastic.

V. RESULTS

The following results were obtained after properly carrying out the experiment in accordance with the above-mentioned approach:

A. Training Results

On training our dataset using the above-mentioned machinelearning and deep learning algorithms, we have obtained the following accuracy on training and test data, respectively, as shown in Table 1. On the basis of observation, we can say that most of the algorithms have high accuracy on training databut show relatively lower accuracy when compared to results with test data.

As we know, accuracy with test data should be considered when choosing the best algorithm because accuracywith train data only indicates how well the training data is fit to our model, whereas accuracy with test data indicates the model's ability to predict on new inputs. As a result, basedon observations, the deep learning-based GloVe model, with anaccuracy of 82.55 percent, which uses the bi-directional LSTMalgorithm as one of its layers within the designed model, is chosen to be the best model among all.

TABLE I ACCURACIES

Models	training data	test data
Multinomial Naive Bayes	99.045	80.86
Stochastic Gradient Descent	99.550	80.906
KNeighbors	74.411	72.932
Decision Tree	53.318	54.148
Logistic Regression	99.810	80.3555
Support Vector	100.0	80.114
Random Forest	100.0	76.818
word2vec(bi-LSTM and bi-GRU)	99.71	79.478
Glove(bi-LSTM)	98.277	82.541

B. Making Predictions Using Completed Project

Once the best model was found, we used that model to create our prediction system, and the results of the completed project are shown below in figure 2 and 3.

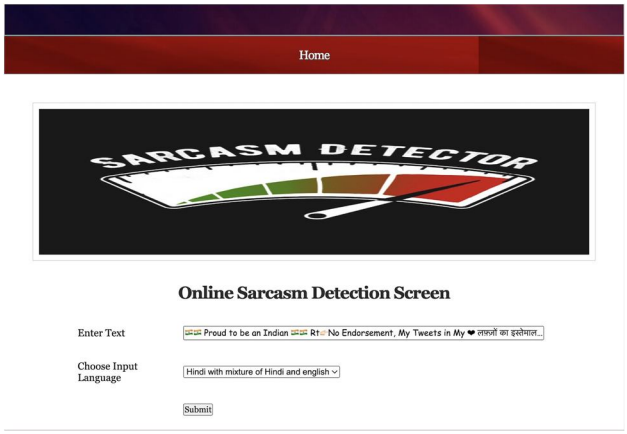


Fig. 2. The home page

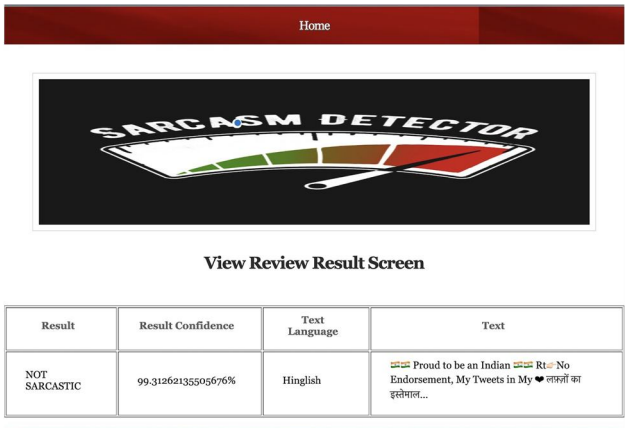


Fig. 3. The results page

VI. CONCLUSION

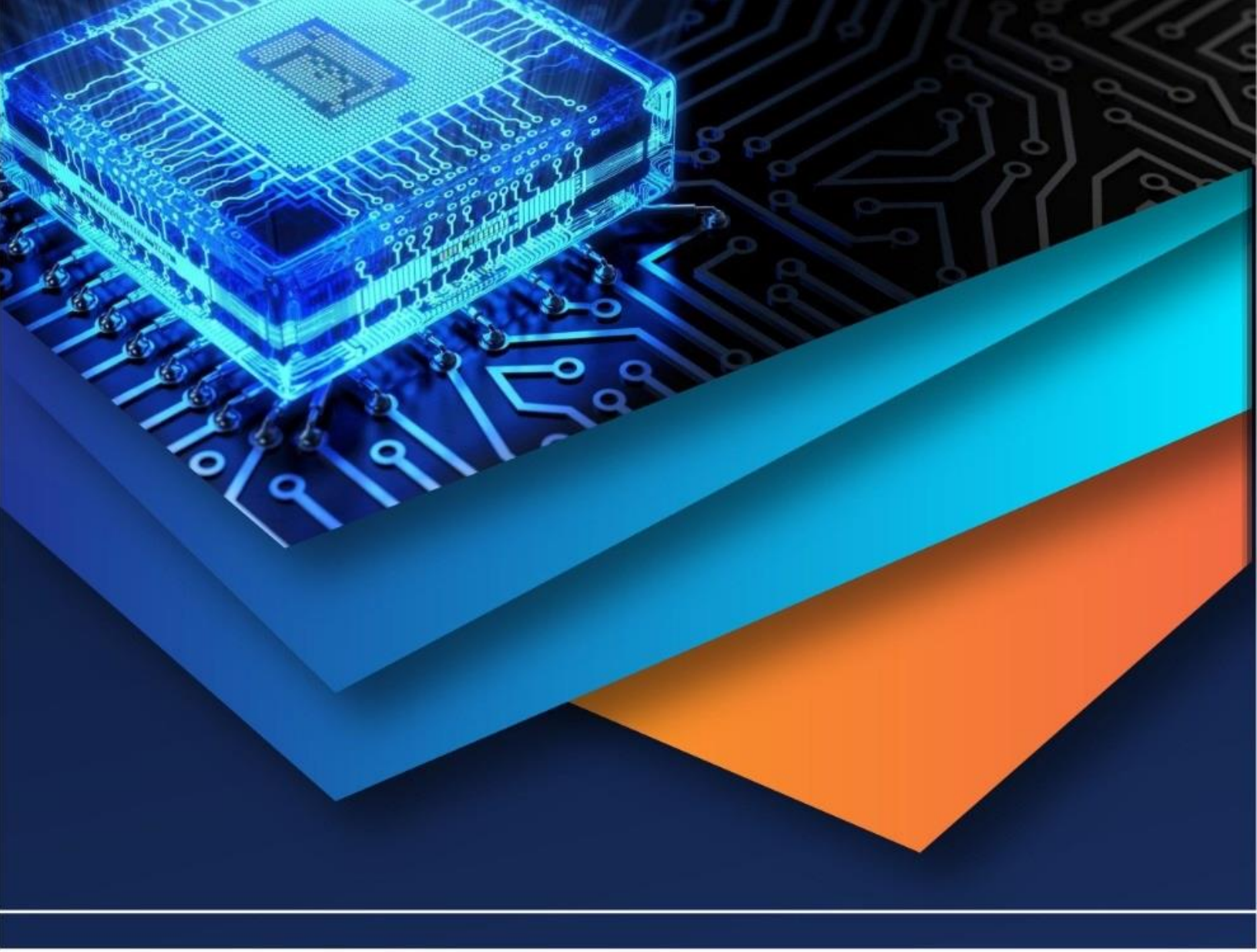
People nowadays frequently communicate their views and opinions through social media, which has paved the way for the development of tasks like opinion mining and various sentiment analysis. The sarcastic element in the sentences makes it increasingly harder to understand their overall meaning, necessitating careful processing and study.

In order to address the issue of sarcasm detection, we evaluated the effectiveness of various deep learning models that use generated word embeddings as their input. Moreover, we've included a function that lets users look for sarcasm in any type of language they wish to search for.

Overall, while sarcasm detection remains a challenging task, recent advances in NLP techniques and data resources have opened up new avenues for research and development. With further refinement and testing, these methods have the potential to significantly improve our ability to detect sarcasmin natural language texts.

REFERENCES

[1] Aditya Joshi, Pushpak Bhattacharyya, and Mark J. Car- man. 2017. Automatic sarcasm detection: A survey. ACM Comput. Surv., 50(5).
[2] Sakshi Gupta, Piyush Bansal, and Radhika Mamidi. 2016. Resource creation for hindi-english code mixed social media text.
[3] David Bamman and Noah Smith. 2015. Contextualized sarcasm detec- tion on twitter.
[4] S.K. Bharti, B. Vachha, R.K. Pradhan, K.S. Babu, S.K. Jena. Sarcastic sen- timent detection in tweets streamed in real time: a big data approach. In Digital Communications and Networks, Volume 2, Issue 3,2016, Pages 108- 121 (2016).
[5] A. G. Prasad, S. Sanjana, S. M. Bhat, and B. S. Harish, "Senti- ment analysis for sarcasm detection on streaming short text data," in 2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA), pp. 1–5, London, UK, 2017.
[6] Pooja Deshmukh, Sarika Solanke. 2017. Review Paper: Sarcasm De- tection and Observing User Behavioral. In International Journal of Computer Applications 166(9):39-41, May 2017.
[7] Amitava Das, Bjo rn Gamba ck. Code-Mixing in Social Media Text:The Last Language Identification Frontier?. In TAL 54: 41-64(2013).
[8] Toma s Pta cek, Ivan Habernal, Jun Hong, Toma s Hercig. Sarcasm Detection on Czech and English Twitter. In COLING(2014).
[9] Bosco, V. Patti, A. Bolioli. Developing Corpora for Sentiment Analysis: The Case of Irony and Senti-TUT. In IEEE Intelligent Systems,vol. 28, no. 2, pp. 55-63, March-April 2013.
[10] Semi-Supervised Recognition of Sarcastic Sentences in Twitter and Amazon D. Davidov, O. Tsur, A. Rappoport Institute of Computer Science The Hebrew University Jerusalem, Israel Proceedings of the Fourteenth Conference on Computational Natural Language Learning, pages 107–116, Uppsala, Sweden, 15-16 July 2010.
[11] Recognition of Sarcasm in Tweets Based on Concept Level Sentiment Analysis and Supervised Learning Approaches P. Tungthamthiti K. Shirai M. Mohd
[12] Identifying Sarcasm in Twitter: A Closer Look R. González-Ibáñez S. Muresan N. Wacholder HLT '11 Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Lan- guage Technologies: Volume 2 81-586 Association for Computational Linguistics Stroudsburg, PA, USA ©2011
[13] S.K. Bharti, B. Vachha, R.K. Pradhan, K.S. Babu, S.K. Jena. Sarcastic sen- timent detection in tweets streamed in real time: a big data approach. In Digital Communications and Networks, Volume 2, Issue 3,2016, Pages 108- 121 (2016).
[14] M. Bouazizi, T. Otsuki Ohtsuki. A Pattern-Based Approach for SarcasmDe- tection on Twitter. In IEEE Access, vol. 4, pp. 5477-5488, 2016.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY