

Methodologies for Sarcasm Detection on Online Social Media

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE AWARD OF THE DEGREE

OF

Master of Technology (M.Tech)

IN

Computer Science and Engineering (CSE)

Submitted by

Govind Narayan Jha (2K20/CSE/09)

Under the Supervision of

Dr. Aruna Bhat

Associate Professor



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY, DELHI**

MAY, 2022


DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
NEW DELHI

DECLARATION

“I, Govind Narayan Jha, student of M. Tech. (Computer Science and Engineering) certify that the project titled “Methodologies for Sarcasm Detection on Online Social Media” that is presented by me to the Department of Computer Science & Engineering, Delhi Technological University, in substantial fulfilment of the requirement for the certificate of the degree of Master of Technology is an authentic research and it has not been reproduced from any origin without appropriate references. This research has not historically been the basis for the granting of any Scholar, Degree, Diploma or other equivalent designation or acknowledgement.”

Place: New Delhi

Date: [30/05/2022](#)


Govind Narayan Jha
(2K20/CSE/09)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
NEW DELHI

CERTIFICATE

This is to assure that the Project Dissertation titled “Methodologies for Sarcasm Detection on Online Social Media” which is submitted by Govind Narayan Jha of Computer Engineering, Delhi Technological University Delhi in partial fulfilment of the requirement for the certificate of the degree of Master of Technology is a transcript of the project work performed by the student under my supervision. To the best of my knowledge, this work has not been submitted in part or whole for any Degree or Diploma to this University or anywhere.

Place: New Delhi

Date : [30/05/2022](#)



Dr. Aruna Bhat
Associate Professor
Department of CSE
DTU

ACKNOWLEDGEMENT

I owe a huge debt of gratitude to my project guide, Dr. Aruna Bhat, Associate Professor, Department of Computer Science and Engineering, Delhi Technological University, Delhi, for offering essential advice and serving as a steady source of inspiration throughout my study. I shall be eternally grateful to her for her unwavering support and encouragement.

I am also grateful to Mr. Raju Kumar, who has guided and helped me during the entire process by providing me with valuable inputs and feedback on my work.



Govind Narayan Jha
(2K20/CSE/09)

ABSTRACT

Sarcasm Detection is procedure to identify the phrases that express a meaning contrary to what it really wants to express. The metaphorical nature of sarcasm presents a significant difficulty for sentiment analysis systems based on emotion detection. Detecting sarcasm on online forums is a very specific topic in natural language processing (NLP), a sort of sentiment analysis that focuses on recognising sarcasm rather than discovering a perception over the whole domain. Sarcasm identification and sentiment analysis vary by a hair's breadth. In NLP, sarcasm recognition is a rather specific research topic. As a consequence, the objective is to figure out if a script or phrase is sarcastic or not. In our previous survey we have discussed and compared the approaches to sarcasm detection. This research paper is focused on how elegantly to analyse the large datasets without losing the efficiency of the Deep Learning models. In recent years there is significant increase in the social media users that is why identifying sarcasm is of great concern. Sarcastic remarks in the form of tweets typically include positive phrases that symbolise bad or unpleasant attributes, therefore recognizing sarcasm on social media has gotten a lot of attention recently. In this paper we implement Knowledge distillation approach to detect sarcasm in news headline. Proposed methodology reduces the complexity of the deep learning neural network by retaining the accuracy.

CONTENTS

DECLARATION	ii
CERTIFICATE	iii
ACKNOWLEDGMENT	iv
ABSTRACT	v
LIST OF FIGURES	viii
LIST OF ABBREVIATIONS	ix
LIST OF GRAPHS	x
CHAPTER 1: INTRODUCTION	1
1.1 Problem Statement	1
1.2 Overview	1
CHAPTER 2: LITERATURE REVIEW	3
CHAPTER 3: BACKGROUND	6
3.1 Collecting Dataset	6
3.2 Data Pre-Processing	6
3.3 Tokenization	6
3.3.1 Stemming	7
3.3.2 Lemmatization	7
3.4 Feature Extraction	8
3.5 Feature Selection	9
3.6 Classification Approaches	9
3.6.1 Rule Based Approach	10
3.6.2 Lexical Approach	10
3.6.3 Machine Learning Methods	10
3.6.4 Deep Learning Methods	14

CHAPTER 4: PROPOSED METHODOLOGY	20
4.1 Knowledge Distillation	20
4.2 Datasets Used	21
4.2 Steps of Knowledge Distillation	22
4.3 DistilBERT MODEL	23
4.4 Mathematical Illustration of Proposed Approach	25
CHAPTER 5: RESULTS	27
CHAPTER 6: CONCLUSION & FUTURE SCOPE	29
REFERENCES	30
LIST OF PUBLICATIONS	33

LIST OF FIGURES

Fig. 1	Tokenization	7
Fig. 2	Stemming vs Lemmatization	8
Fig. 3	Feature Selection	9
Fig. 4	Decision Tree	11
Fig. 5	Random Forest	12
Fig. 6	Support Vector Machine	13
Fig. 7	Recurrent Neural Network (RNN)	14
Fig. 8	Convolution Layer	16
Fig. 9	LSTM gates	17
Fig. 10	Attention Mechanism	18
Fig. 11	Knowledge Distillation Framework	21
Fig. 12	Knowledge Distillation	22
Fig. 13	Knowledge Distillation Flowchart	23
Fig. 14	Proposed Model Implementation	24

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
DL	Deep Learning
ML	Machine Learning
SVM	Support Vector Machine
NLP	Natural Language Processing
LSTM	Long Short Term Memory
BERT	Bidirectional Encoder Representations from Transformers
RoBERTa	Robustly Optimized BERT pre-training approach

LIST OF GRAPHS

Graph 1.	Outcomes of Feature Extraction	3
Graph 2.	Result Comparison of Previous Methodologies	5

Chapter 1

INTRODUCTION

1.1 Problem Statement

Sarcasm Detection is the procedure to identify the phrases that express a meaning contrary to what it really wants to express. The metaphorical nature of sarcasm presents a significant difficulty for sentiment analysis systems based on emotion detection. As a rule, Sentiment analysis expects to determine the speaker's or author's point of view as a specific point or tone of a statement.

1.2 Overview

Sentiment Analysis is a data mining goal for sentiment categorization of customer reviews that are accessible in text format. Sarcasm is a type of speech that uses positive language to describe a person's negative sentiments. Sarcasm labelling in typescript is critical for Natural Language Processing to avoid the misconception of sarcastic statements being literal ones. Finding these sorts of sarcastic phrases is difficult for both people and machines. Sarcasm has a stronger impact on the efficiency of Sentiment Analysis models that are impacted by deceptive attitudes, which are frequently seen in the sardonic category. People may communicate their thoughts and feelings via text, emoticons, and photographs on the vast platform of social media. Many organizations use this information to better understand how people feel about products, movies, and political events. The enormous problem for the Sentiment Analysis task is to correctly identify remarks into three categories: positive, negative, and neutral. Sarcasm leads to a misinterpretation when it comes to determining the polarity of a statement. As a result, detecting sarcasm has become a difficult problem in the Sentiment Analysis assignment. Context and expression are crucial in sarcasm. "It's a fantastic sensation to squander my

valuable hours in traffic," for example. The positive vocable 'awesome' is used to indicate a negative sensation of losing time in traffic in the above-mentioned comment.

With the rise in popularity of social media, detecting sarcasm will become much more difficult. This kind of speech was common in micro blogs and social media, making it impossible for authors to individually assess each text, leading to the development of a program to spot sarcasm. Sarcasm Detection is a subset of NLP that deals with text-based comedy analysis. Rule-based approaches, as well as for analytic methods, have been employed in the past to predict sarcasm using (i) lexical features (ii) pragmatic features (iii) the presence of polarity shifts in moods, punctuations, and so on.

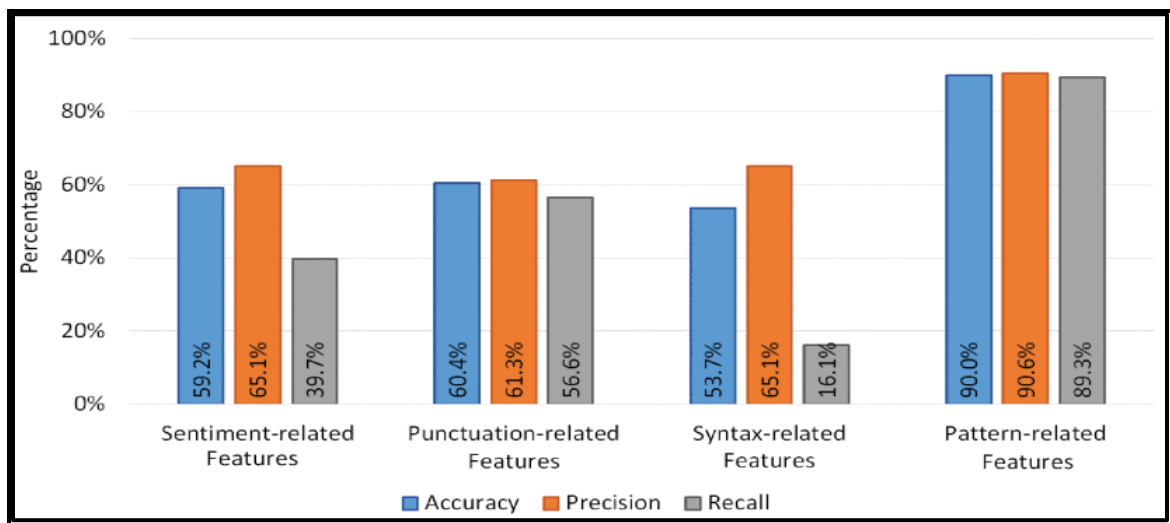
In Natural Language Processing challenges, all of the DL frameworks appear to provide intriguing results. Despite using handmade features, a DL neural network approach learns the essential features without human influence.

Chapter 2

LITERAURE REVIEW

Sarcasm detection is a difficult problem of NLP. Recent researches on sarcasm detection shows machine learning and deep learning approaches performs better in terms of accuracy of the model is concern. Instead of employing handmade features, a deep neural network may learn essential features automatically. In numerous natural language processing (NLP) tasks DL models have excellent performance. As demonstrated in machine translation, phrase summarising, and boosting reading comprehension, the attention mechanism aids deep learning model performance.

In [1] A.Vaswani et al. proposed a multihead attention algorithm [2] in order to extract the exact meaning of tokens in various contexts. A Rajadesingan et al.[3] proposed behavioural modelling methodology to identify sarcastic comments on twitter datasets. Under this algorithm user's previous tweets were analysed. Now using the result of the analysis a behavioural framework is designed for sarcasm detection. In [4] author implemented pattern based features to detect sarcasm. These features are extracted from each tweets which include syntax and semantic features, sentimental features, punctuation related features and pattern related features. Graph 1 [4] depicts the outcome of each features.



Graph 1. Outcomes of Feature Extraction

Machine learning methodologies are used for detecting sarcastic phrases automatically. Saurabh Porwal et al. [5] plan to employ a recurrent neural network (RNN) model since it naturally retrieves information needed for machine learning techniques. This model, in addition to the recurrent neural network, leverages tensorflow's LSTM cells to gather syntactic and semantic information from tweets in order to identify sarcasm.

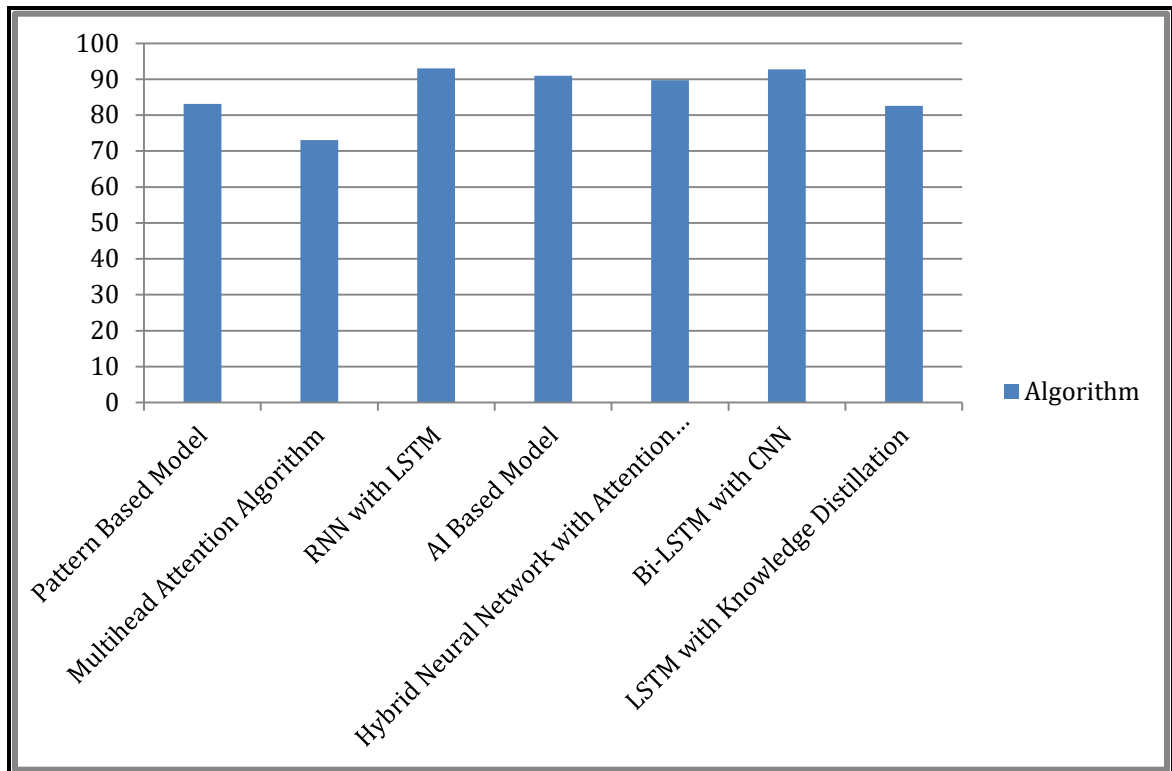
In [6] Abdullah Y. Maaad et al. proposed AI based technique for sarcasm detection on Arabic tweets. Two Arabic twitter datasets are used in this study. For the research work, binary and multiclass issues are presented for each case study. The AraBERT classifier has the highest accuracy for misogyny detection, with 91.0% for classification and 89.0% for multiple class scenarios. The AraBERT also has the best performance for sarcasm recognition, with an accuracy of 88 percent for classification and 77.0% for multiple class. With AraBERT as a suggestion, the proposed technique appears to be effective in identifying sexism and sarcasm in social media networks. Recent researches on sarcasm detection significantly use twitter datasets but the twitter datasets are noisy. In [7] R Misra and P Arora suggested a new dataset including media stories from a funny news site and a legitimate news source to address the shortcomings. The authors used a composite Neural Network architecture with an attention mechanism to figure out what makes words sarcastic and show that the recommended strategy increases accuracy by 5% over the benchmark in testing.

Jain, D., Kumar, A., Garg, G [8] performed real time sarcasm detection using deep learning approach. The effectiveness of the recommended technique is evaluated using real-time datasets from hot political posts and entertainment posts on social media. From the dataset, 50% sarcastic and 50% non-sarcastic bilingual Hindi as well as English tweets.

Aruna Bhat et al. [9] worked on twitter datasets of COVID period when majority of people are engaged in social media platforms. The main concern at this time is the spread of hate comments and thoughts. This paper focuses on detecting those hate speech from the twitter datasets. In this paper author proposed the concept of Knowledge Distillation [10] to improve the result of LSTM model. Concept of the proposed method based on Teacher and Student model. The basic idea behind this approach is to feed the knowledge of teacher's model into the student model. The accuracy of the student model without

knowledge transfer of teacher model is 75.4% and by including knowledge of teacher model accuracy increased to 82.6%.

Graph 2 represents the accuracy of significant methodologies for sarcasm detection discussed so far.



Graph 2. Result Comparison of Previous Methodologies

Chapter 3

BACKGROUND

In this section we will discuss the basic procedure to carry out the process of sarcasm detection.

3.1 Collecting Dataset

Data acquisition is still a crucial phase in any model. There are two methods for gathering data for detecting sarcasm.:

- (a) API (Application Programming Interface) [11]
- (b) Accessible Datasets like
 - MUStARD (Multi-modal Sarcasm Detection dataset)
 - SARC (Self annotated Reddit Corpus)
 - SemEval (Semantic Evaluation) dataset
 - IAC (Internet argument Corpus)
 - News Headline

3.2 Data Pre-processing

After collecting the datasets now we will perform pre-processing which basically removes noise from the raw data collected in above step.. This process includes Stop-word eradication, tokenization, stemming, and lemmatization of data tokens for NLP tasks.

3.3 Tokenization

Using a tokenizer [12], un-processed data and language processing text are segmented into snippets of data that may be recognized distinct components. The occurrence of tokens in a text can be used to generate a matrix that reflects the text. An unprocessed phrase (text document) is turned into a scientific data structure suitable for learning algorithms in a matter of seconds. They can also be utilised to direct a machine's beneficial actions and reactions. They might potentially be used as characteristics in a

machine learning network to prompt more complex decisions or actions. Tokenization of text can be done on basis of rules, punctuation, white spaces as shown in Fig. 1

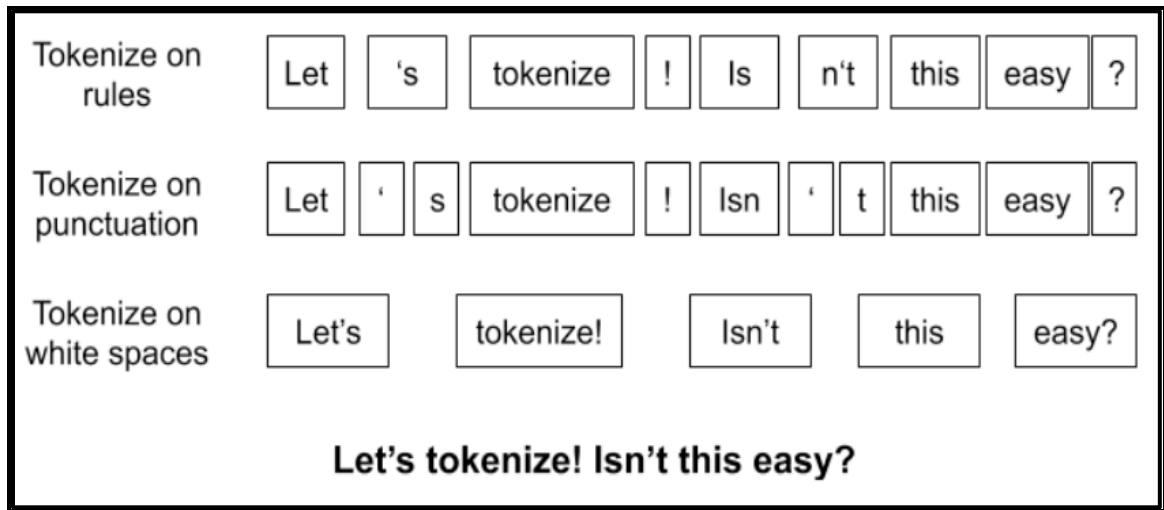


Fig. 1

3.3.1 Stemming

Stemming is the way of eliminating a phrase to its word stem, which usually refers to the beginning or base of words known as a lemma [13]. Stemming is beneficial to both linguistic form understanding (NLU) and NLP . Stemming is performed on the text corpus in order to identify the each word from its base form. An example of Stemming is illustrated in fig. 3.

3.3.2 Lemmatization

Lemmatization [14] is a concept that refers to utilising a vocabulary and grammatical evaluation of words to remove inflectional endings and restore the core or glossary form of a word, also defined as the lemma. Algorithms of lemmatization refers to a predefined dictionary to understand the exact meaning of the word in the context it is used and then reduce it to its root word. An example of Lemmatization is illustrated in fig. 3.

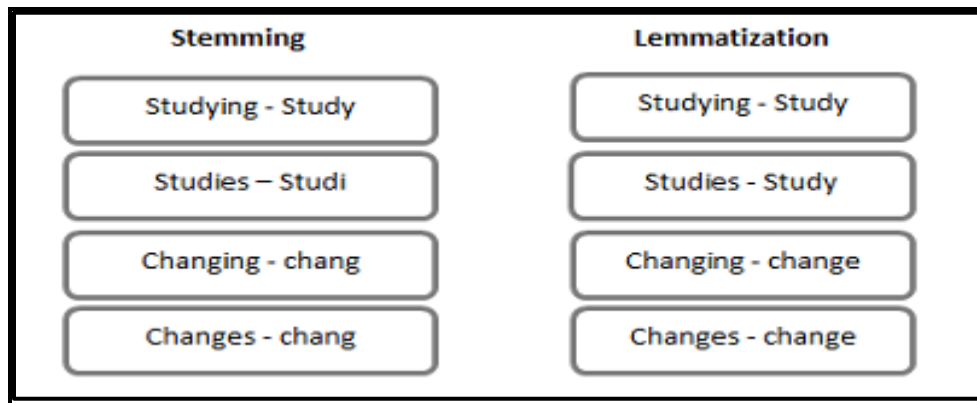


Fig. 2

3.4 Features Extraction

After removing noise from the data, next step is to extract features from the datasets.

Various features for Detecting Sarcasm in text are [15]

- Lexical Features
 - Uni-gram
 - Bi-gram
 - N-gram
- Hyperbole Features
 - Punctuation Mark
 - Quotes
 - Interjection
- Pragmatic Features
 - Emoji / Emoticons
 - Capitalization
 - Replies
- Syntactic Features
 - PoS- grams
- Pattern Based Features
 - Appearance Frequency of Words

3.5 Feature Selection

The technique of picking a portion of the phrases in the training set and using just this portion as attributes in text classification is known as feature selection[16]. The objective of feature selection is twofold. First, it improves the efficiency of training and executing a classifier by reducing the quantity of the applicable vocabulary. This is especially important for classifiers that are more difficult to train than Naive Bayes. Second, by removing noisy characteristics, feature selection frequently improves classification performance. Fig. 4 shows various methods of feature selection.

Chi-square and Mutual information (MI) strategies are popular feature selection algorithms.[17].

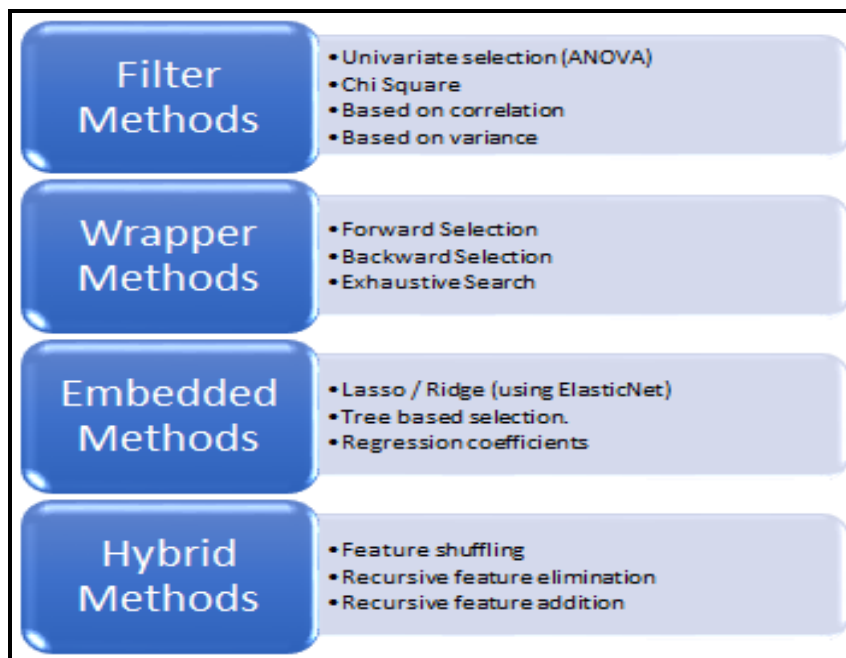


Fig. 3

3.6 Classification Approaches

The difficulty of analysing the text corpus as sarcastic or not is a dual classification problem. Machine Learning, Deep Learning, and their hybrid grouping approaches have been used in previous investigations for categorization purposes. Various classification approaches are as follows [18]

3.6.1 Rule-Based Approach

It includes syntactic, pragmatic, semantic and prosodic approaches.

- The syntactic technique is concerned with the principles for constructing sentences.
- Semantics is concerned with the meaning of tokens created.
- The presence of punctuation marks is referred to as the pragmatic technique.
- The term "prosodic" relates to a person's personality and cadence.

3.6.2 Lexical Approach

Opinion phrases are utilized to represent the sentiment in this approach. It is divided into two groups corpus-based and dictionary-based.

Corpus-Based

It is based on a syntactic pattern - a set of opinion words that appear together in a certain context.

Dictionary Based

The collection of opinion words is gathered in this method.

The collection size is then extended by using a well-known repository such as WordNet to locate similar words.

3.6.3 Machine Learning Methods

ML models are used to automatically assess attitudes based on learned characteristics from datasets. It includes basically four methods as discussed below[20]

Decision Tree

It is the special algorithm for prediction and classification problems. It consists of tree like structure. The population starts dividing from root nodes. The nodes which are obtained after the categorization of root nodes are termed as Decision Nodes. When the

nodes cannot be splitted further then at that point the nodes are termed as Leaf node.
Concept of Decision tree is illustrated in Fig.4

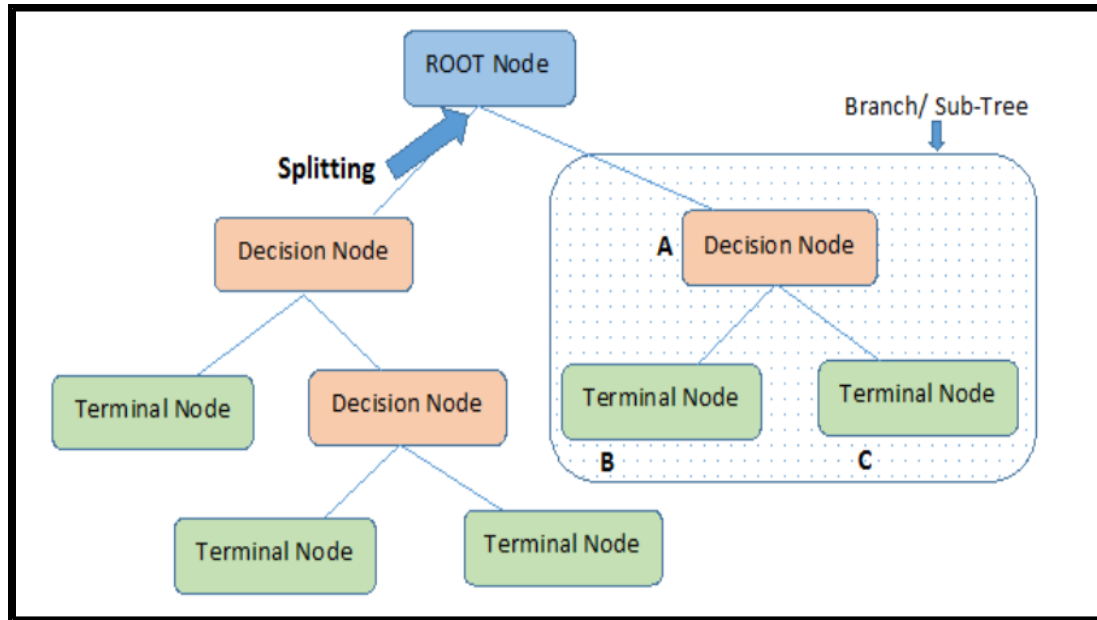


Fig. 4

Naive Bayes

The Naive Bayes algorithm determines whether a data item corresponds in a specific category or not. It can be used in text processing to categorize words or sentences as belonging to a predefined label. Based on the Bayes rule, this is a basic (naive) categorization approach. It works on the basis of a very basic word embedding.

Consider the following scenario: we have two classes (positive and negative), and the input is a text that represents a movie review. If the review was favourable or bad, please let us know. As a result, we may have a bag of good and bad words. To identify the document as positive or negative, we may count how many times each of those phrases appears in the content.

Random Forest

It's is a machine learning methodology that may be used to solve issues such as regression and prediction. It makes use of supervised techniques, which are a method for addressing problems requiring several classes. There are several decision trees in a random forest approach. Classifier or bootstrap accumulation are used to train the 'forest' created by the random forest technique (Fig. 5). Bagging is a machine learning approach that uses an ensemble meta-algorithm to optimise a model.

- Based on the decision tree estimations, the (random forest) approach decides the result. It generates estimates by combining the output of several trees. The quality of the outcome improves as the number of trees rises. The following are some of the benefits of using this algorithm:
- Accuracy is High
- Missing Data are handled effectively
- Eradicate the overfitting issues

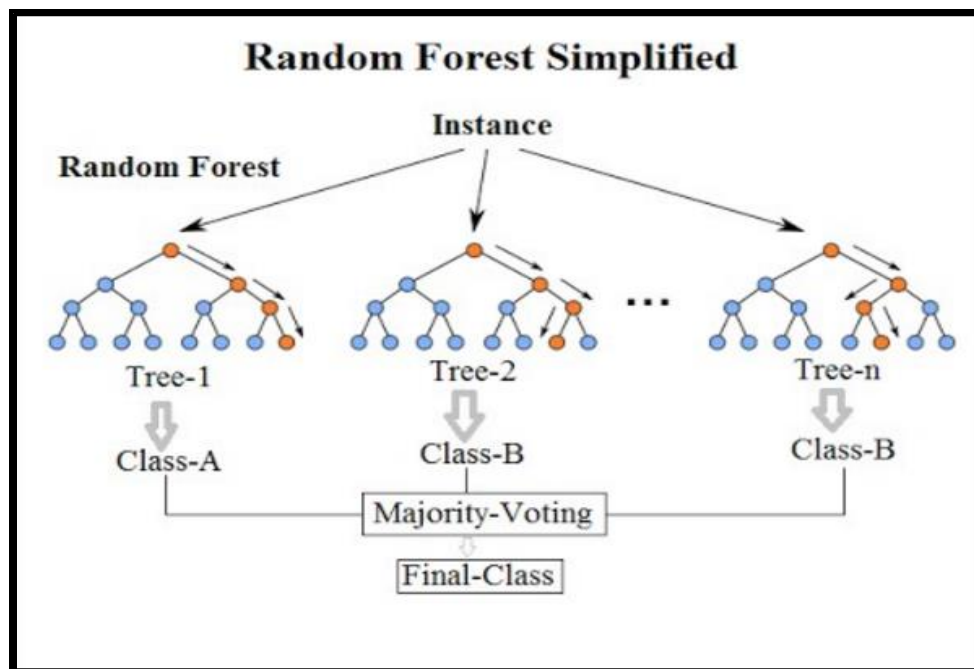


Fig. 5

Support Vector Machine

SVM model in more dimensions is effectively a hyperplane representation of class labels. In order to minimise the error, SVM will create the hyperplane iteratively. In order to find the biggest marginal hyperplane, SVM divides datasets into groups. The purpose of this method is to find a support vectors in N-dimensional space that differentiates between data points ($N = \text{the number of characteristics}$).

Important terms in SVM are (Fig. 6)

Support vectors are the data points that are nearest to the hyperplane.

Hyperplane- It is a decision plane or space.

Margin- It's the gap between two distinct categorization

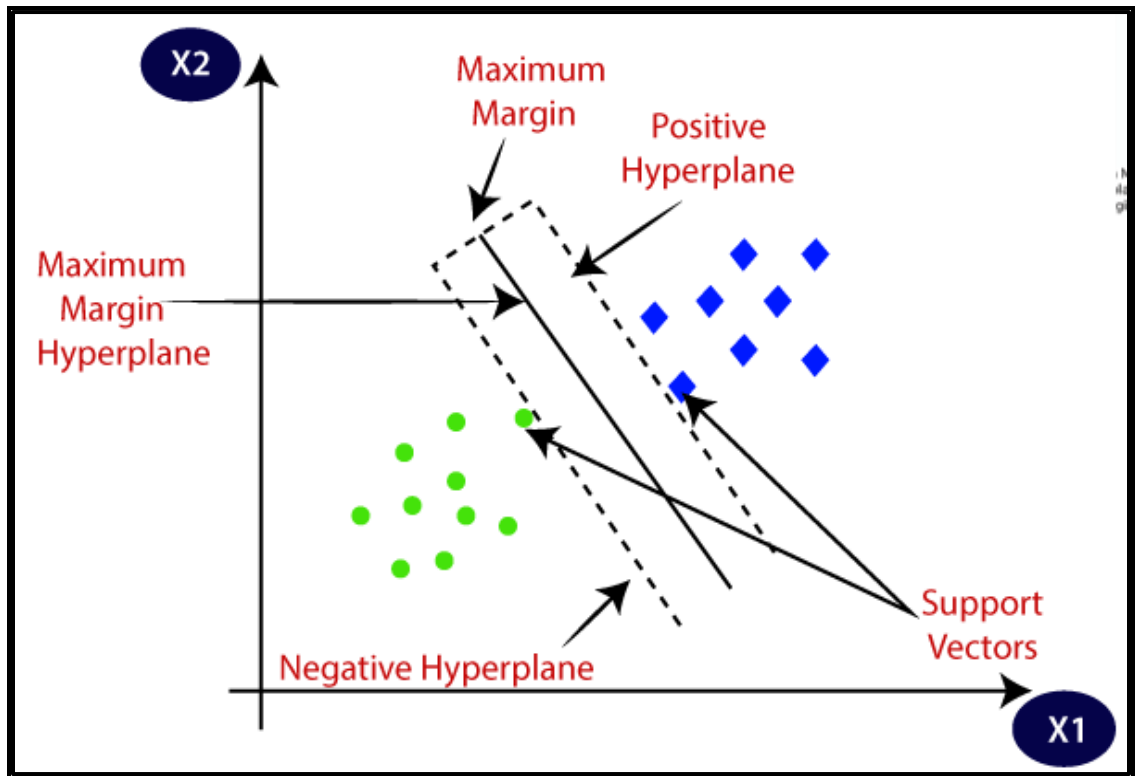


Fig. 6

3.6.4 Deep Learning Methods

Deep learning methods were utilized to find local and irreversible position anomalies in the comments. Various deep learning methods are discussed below.

Recurrent Neural Network (RNN)

RNNs are particularly well suited to challenges in which the sequence is more essential than the specific elements. An RNN is simply a fully connected neural network with some of its layers refactored into a loop. A repetition over the concatenation or composition of two inputs, a matrix multiplication, and a non-linear function is often used in that loop. RNNs contain a kind of internal memory that allows earlier inputs to influence future predictions. If you know what the preceding words were, it's much easier to anticipate the following word in a phrase with greater precision.

Each word in a phrase is treated as a separate input at time 't,' and the activation value at time 't-1' is likewise used as an input in parallel to the input at time 't.' This is illustrated in Fig. 7.

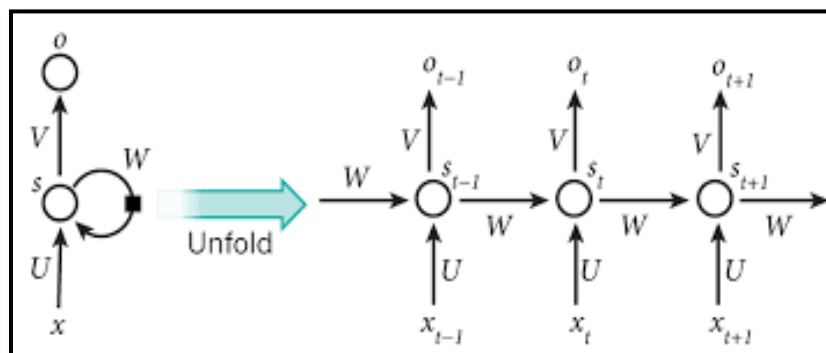


Fig. 7

Convolutional Neural Network (CNN)

A convolutional neural network is a type of deep neural network that is most typically used to analyse visual images in deep learning. Multilayer perceptrons are regularised variants of CNNs. These networks' "total connectivity" renders them prone to data overfitting. Regularization, or avoiding overfitting, may be performed in a number of approaches, including penalising characteristics during training or eliminating connections. CNN's employ a distinct ways to regularisation: they take use of the hierarchical structure in data and use smaller and simpler sequences imprinted in their filters to generate patterns of rising complexity. As a consequence, CNN's are at the lowest end of the connectivity and complexity scale.

CNN Layers

CNN is comprised of basically three layers (Fig 8)

Convolution Layer

Convolutional layers convolve the input before sending the results to the next layer. In brain activity, this is similar to a neuron's reply to a single stimulus. Each convolutional neuron is exclusively responsible for processing data for the perceptron to which it has been allocated. While fully connected feed forward artificial neural network may be used to train the classifier and categorise data, they are not ideal for larger inputs such as high-resolution photographs.

Pooling Layer

After a Convolutional Layer, a Pooling Layer is frequently added. The main function of this level is to cut the volume of the identification stage feature map in attempt to lessen computation costs. This is achieved by minimising layer connections and operating independently on each feature map. Based on the mode used, there are several types of pooling operations.

The feature map in Max Pooling yields the largest attribute. Average Pooling is used to compute the sum of the factors in a given sized Image segment. Sum Pooling is a method

of calculating the entire amount of the components in a given section. Typically, the Pooling Layer is utilised to link the Convolutional Layer to the fully connected level.

Fully Connected Layer

In completely linked layers, every synapse in one layer is connected to every synapse layer of the network. It functions similarly to a multilayer perceptron neural network (MLP). The flattened matrix runs through a fully connected layer to classify the graphics. In fully connected layers, the neuron applies a linear change to the input vector using a weights matrix. The result is then subjected to a non-linear transformation using a non-linear activation function.

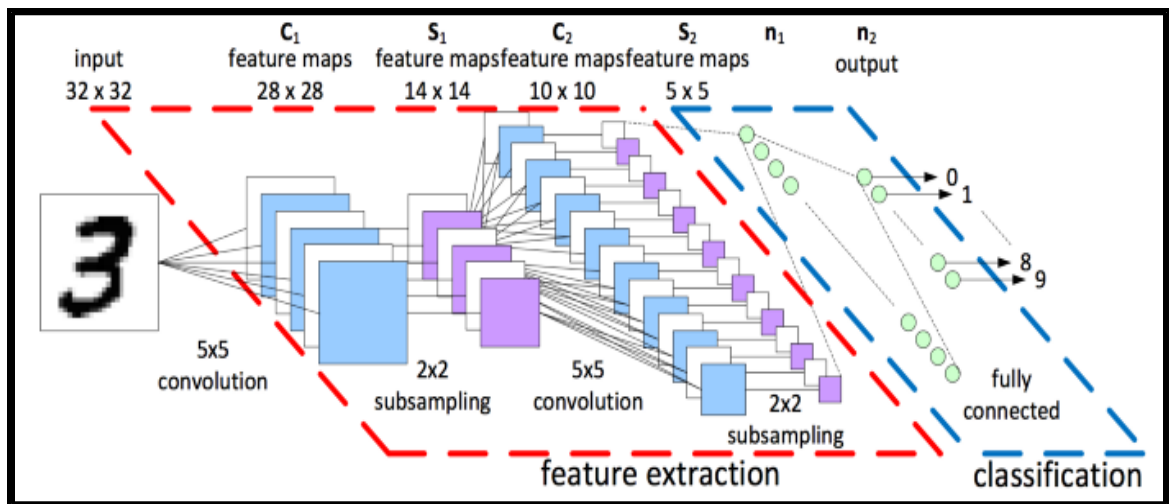


Fig. 8

Long-Short Term Memory (LSTM)

LSTM is a kind of recurrent neural network that beats traditional recurrent neural networks when it comes to memory. When it comes to learning particular patterns, LSTMs [21] considerably outperform other algorithms. The important facts are maintained while the extraneous data is discarded in each cell as the LSTM, like any other NN, passes through each layer.

LSTM consists of three main gates (Fig. 9)

- **FORGET Gate:-** This gate responsible for deciding which information should be preserved and which should be rejected when computing cell state. h_{t-1} conveys information from the prior hidden state (previous cell), whereas x_t represents knowledge from the existing cell. These two inputs are sent to the Forget gate. They're passed through a sigmoid function, with the ones that lean to 0 being destroyed and the remainder being used to figure out the cell state.
- **INPUT Gate:-** This Gate adjusts the cell's state and decides which data is necessary and which is not. Like the previous gate discussed above, the input gate assists in the finding of critical data and the storing of important content in the storage. The sigmoid and tanh functions are used to process the inputs h_{t-1} and x_t , respectively. The tanh function is used to manage the network and reduce bias.
- **OUTPUT Gate:-** It is the last gate, which is responsible for deciding what the next hidden state will be. The parameters h_{t-1} and x_t are used to invoke a sigmoid function. To detect what data is being stored in the hidden state, the newly adjusted cell state is routed through the tanh function and multiplied by the sigmoid output.

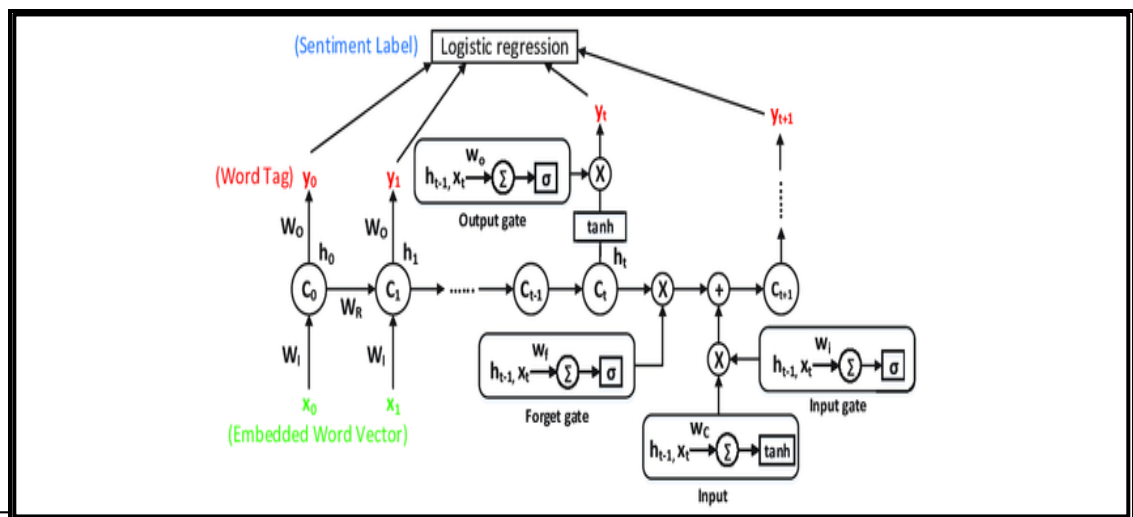


Fig. 9

Attention Mechanism

The attention mechanism [22] is a component of a neural architecture that allows users to seamlessly highlight significant parts of incoming data, which in NLP is often a series of textual components. It can be applied on the image input or its higher-level representation directly. The basic concept underlying focus is to build a weight distribution on the training data, with larger values being assigned to more relevant items. To demonstrate, we will quickly introduce RNN search a typical attention architecture. RNN search was chosen because of its historical relevance as well as its simplicity in comparison to other structures.

In the context of neural networks, attention is an approach that replicates cognitive attention. The approach highlights significant parts of input data while fading out the rest, with the goal of allocating greater computing resources to that small but crucial fraction of the data. The context determines which input component is more important than others, and this is learned via gradient descent and training data.

Attention Mechanism Techniques

- **Dot Product Attention-** It performs dot product between the vectors to calculate attention.
- **Multi-head Attention-** This method is comprised of various attention mechanisms to calculate the overall attention of the network. Fig. 10 illustrates the Bahdanau Attention Mechanism.

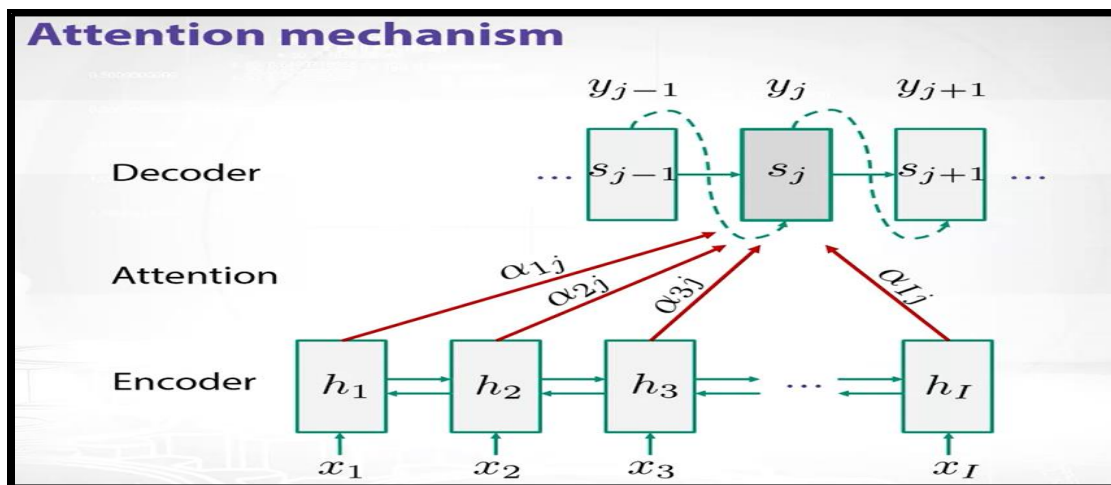


Fig. 10

The step-by-step procedure for using the Attention Mechanism is as follows:-

- Generating Encoder Hidden State

Each element in the input sequence is encoded to yield hidden states.

- Calculating Alignment Scores

Alignment score is calculated among each encoder's hidden state and the previous decoder's hidden state.

- Softmaxing of Alignment Score

- Context Vector Calculation

- Output Decoding

The above-mentioned Context Vector is integrated with the prior decoder results sent out into the Decoder RNN for that time unit.

- Now The operation (steps 2-5) repeats itself for each time step of the decoder until an item is created or the outcome exceeds the given optimum size.

Chapter 4

PROPOSED METHODOLOGY

This chapter illustrates the proposed methodology for detecting sarcasm. We proposed the methodology of DistilBERT on News Headline datasets to classify new headline as sarcastic or not. This approach is developed by implementing Knowledge Distillation on BERT algorithm of deep learning to solve the problem of sarcasm detection. This method is efficient in handling large datasets without affecting the accuracy of the algorithm.

4.1 Knowledge Distillation

Knowledge Distillation is a technique for transferring information from a huge deep network to a compact deep learning model [10]. As a result, knowledge distillation [23] is a technique for compressing the model while keeping quality. The larger network that imparts information is referred to as a Teacher Network, while the smaller network that receives knowledge is referred to as a Student Network [24]. In a variety of applications, neural networks have shown to be quite effective. The size of Neural networks is typically enormous (millions/billions of parameters), necessitating the use of computers with enough memory and processing capacity to train and deploy them. The model must be implemented on systems with little computing power, such as smartphones and external drives, in various applications. In the medical industry, for example, restricted compute power systems are employed in remote places where models must be performed in real-time. It is desirable to have ultra-light and accurate deep learning models in terms of both times (latency) and memory (computation power). However, ultra-light (a few thousand parameters) models may not be accurate enough. This is where Knowledge Distillation comes into play, with the support of the teaching network [25]. Fig. 11 depicts the basic idea of Knowledge Distillation. Let's focus on the process of the approach, first of all, we pre-process the datasets.

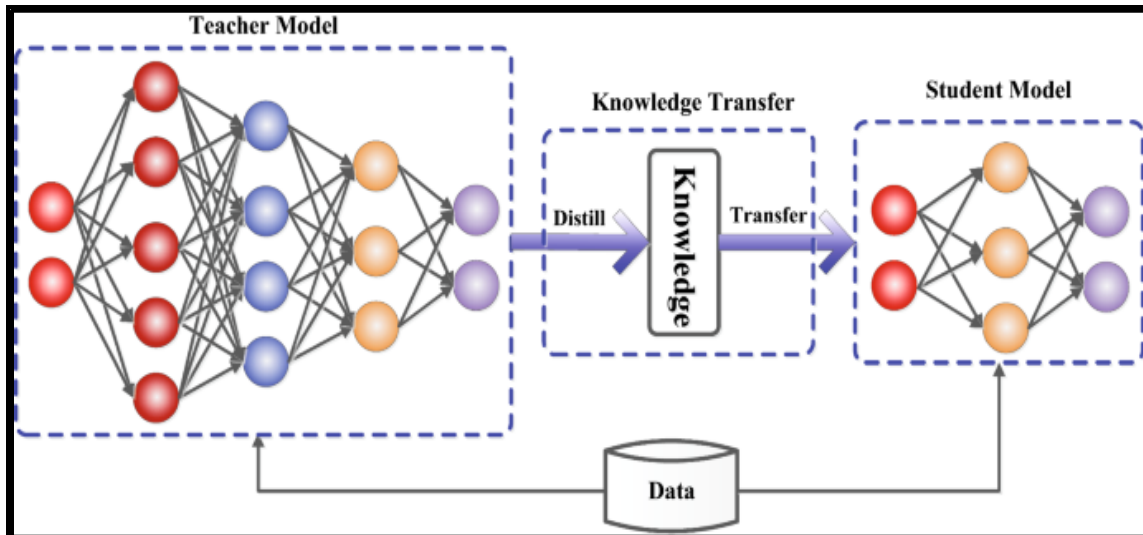


Fig. 11

4.2 Datasets Used

In our proposed method news headline datasets are used. The News Headlines dataset for detecting sarcasm was gathered from two news websites. The Onion's goal is to create satirical representations of current events. Entire sarcastic headlines from the news in short and news in image categories, as well as actual (and non-sarcastic) news headlines from HuffPost, are included in the dataset. News headline datasets give more accurate results as compared to twitter based datasets. Proposed datasets have following advantage over twitter datasets

- Because news headlines are produced by experts in an official way, there are no misspellings or colloquial use. This reduces invariance while improving the probability of discovering pre-trained deep features.
- Importantly, given The Onion's primary goal is to provide funny information, the dataset includes increased labels with much less uncertainty than Twitter datasets.
- Despite tweets that are answers to other tweets, the received media headlines are conscience.

Dataset consist of 3 parameters

- is sarcastic: 1 means sarcastic and 0 means non sarcastic
- news article headline
- link of news article: news article link. It's useful for gathering further facts.

4.3 Steps of Knowledge Distillation

- Declaring Teacher And Student Network

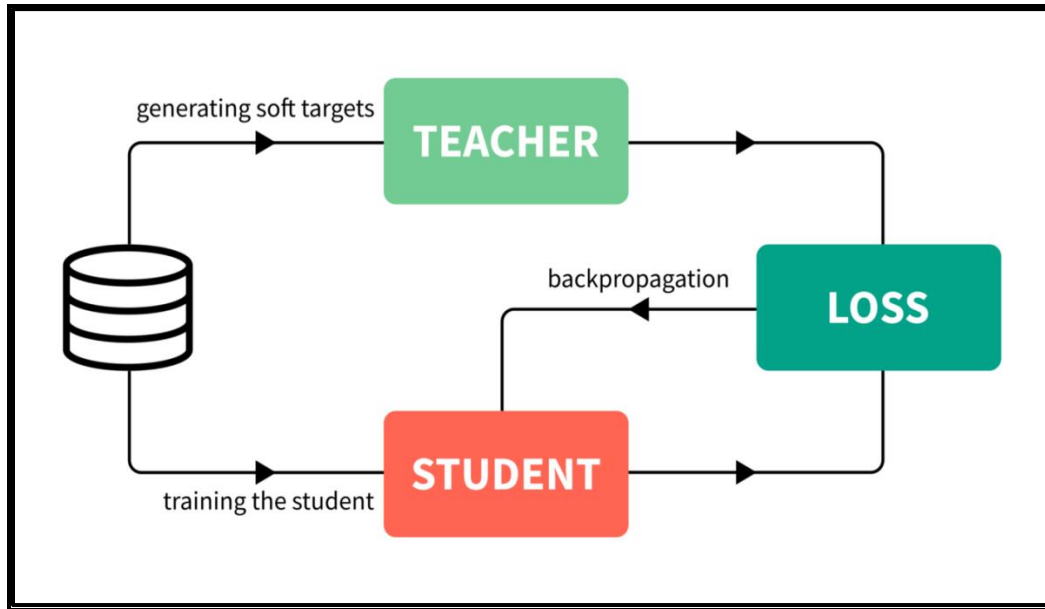


Fig. 12

- Training of entire Teacher network- The teacher network is trained individually at first until it reaches full convergence. In this step, the teacher's network is trained on the entire dataset. This process takes lots of time and memory. Backpropagate mistake in the student node using the teacher network's outcomes and the correspondence direct relation, so that the whole student system may learn to mimic the teacher network's action. Many novel modifications to the classic student-teacher mentioned above have been proposed, such as introducing numerous teachers introducing a teaching assistant (the teacher first teaches the Teaching Assistant, who then teaches the student) [26], and so on.

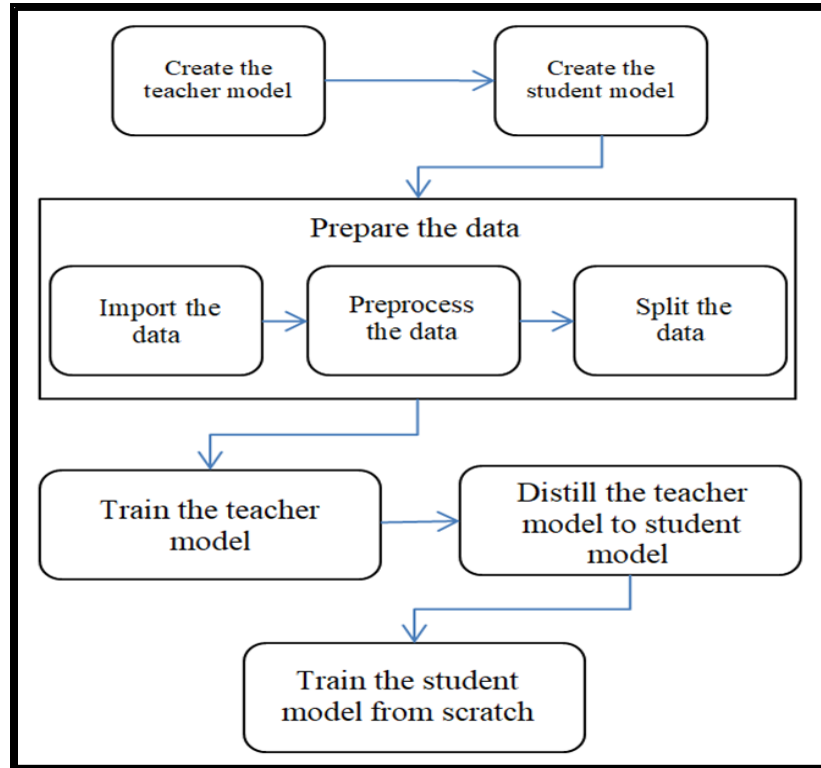


Fig. 13

- Train Student network with Teacher Network- Intelligently train the student network in conjunction with the instructor network. The student network receives training in parallel with the fully qualified instructor network. Both the teacher and student networks are used for advance propagation, whereas the student network is used for backpropagation. Two loss functions have been defined. The student loss and distillation loss functions are two examples.

4.4 DistilBERT MODEL

BERT is a deep learning paradigm for natural language processing that is open source (NLP). BERT is a technology that utilises neighbouring text to help computers understand the meaning of ambiguous phrases in message. The BERT [27] approach was created using Wikipedia text and may be quite well with datasets of lot of queries.

Transformers is a deep learning system in which each output feature is connected to each input parameter, and the prioritisation between them are periodically established based on their connection. BERT is built on Transformers. The use of the BERT model is primarily motivated by the fact that it is bidirectional. As we know from Transformers, BERT stands for **B**idirectional **E**ncoder **R**epresentations from **T**ransformers, and its bidirectionality aids in the simultaneous processing of text that means

- Parallel Processing of Datasets
- Saving the input in order to maintain the correct sequence of text corpus.

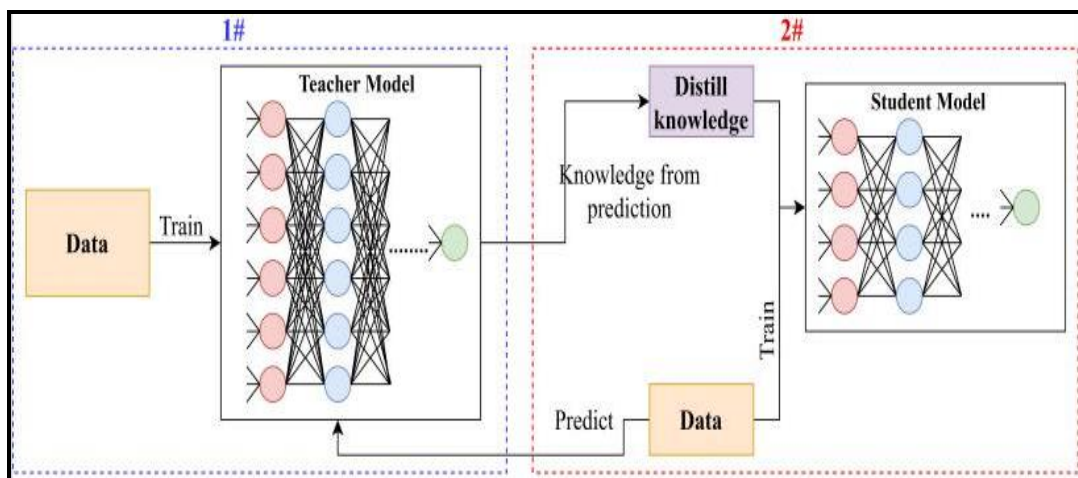


Fig. 14

DistilBERT is very simple and less complex transformer algorithm developed by distilling the BERT base. Fig 14 illustrates the implementation of our proposed methodology. In the figure 1# represent the pre trained teacher model which trained on datasets. After that the learning of teacher model is fed into the student network. This maintains the efficiency of the model but reduces the complexity. The methodology is based on offline distillation in which the teacher model is pre-trained on datasets and after that the knowledge is distilled to the student neural network (2#).

4.5 Mathematical Illustration of Proposed Approach

Initially we define the loss function for teacher and student

$$L_T = H(p, q_T) \quad (1)$$

$$L_{TS} = \alpha * \text{Student Loss} + \text{Distil Loss} \quad (2)$$

where

$$\text{Distillation Loss} = H(q_T, q_S) \quad (3)$$

$$\text{Student Loss} = H(p, q_S) \quad (4)$$

Here,

H-Loss Function;

q_T : Softmax (Z_t / T); q_S : Softmax (Z_s / t)

Z_t and Z_s : pre – Softmax logits

Alpha (α) and Temperature (t) are hyperparameters.

Calculating Softmax

$$S(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}} \quad (5)$$

Applying distillation , Softmax is calculated

$$q_i = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)} \quad (6)$$

Distillation Score is given by

$$DS = \alpha \times (S_S \times T_S) + (1 - \alpha) \times (1 - S_a \times T_a)$$

where S_S and T_S are size of student and teacher model and S_a and T_a are their corresponding accuracy.

The value of Alpha lies between 0 and 1. Hyperparameter temperature can have values 1 to 20. It is observed experimentally that when the student model is small in comparison to teacher model then small the value of temperature gives better performance. When the value of temperature increases, the ensuing soft-label distribution becomes more information-rich, and a tiny model may not be able to capture all of it. We provide a distillation measure for evaluating different knowledge distillation techniques and picking a suitable model from a significant number of student models for execution. The proportion of a student's dimensions to that of the teacher is taken into consideration, as well as the student's accuracy score to that of the teacher. The initial proportion should be as minimal as feasible to get a good size reduction. The second ratio should be as near to 1 as feasible for a distillation procedure to maintain its precision. The student in DistilBERT [28] had the same architecture as BERT and was created with a new triplet loss that incorporated losses from language modelling, distillation, and cosine-distance loss. At last, we distil the teacher model to the student model and after that, we train the student model from scratch. In order to measure the improved results gained by knowledge distillation, we may also train an analogous student model from start without the teacher. When the teacher is trained for 5 complete epochs and the student is focused on this teacher for 3 full epochs, we observed an improved performance when compared to training the identical student prediction model, and even when compared to the teacher. The teacher's precision should be approximately 97.6%, the student who was taught from scratch around 97.6%, and the distilled student around 98.1 %.

Chapter 5

RESULTS

Now we will discuss the outcomes of our proposed approach and compare it with the existing approach. Previous work was majorly performed on Twitter datasets which contain lots of noise in terms of meaningful words. News headline datasets are well structured and have fewer grammatical mistakes which help our model to pre-process the data efficiently and reduces the complexity of evaluation. In the pattern-based approach the feature-based classification is performed to implement the sarcasm detection process. Four features are extracted from twitter datasets and on the basis of that classification is performed. This method gives an accuracy of 83.1%. with 91% precision. DL based algorithms to detect sarcasm also performs efficiently with better results. For example Transformer based approach using recurrent CNN gives accuracy of 82% on SemEval datasets. DL methodology of LSTM followed by CNN gives the accuracy of 93% when implemented on twitter datasets. In R-CNN based approach is based on the pre-trained networks. In this approach initially, the data is processed through the Roberta model. This method is robust and gives an accuracy of 82%.

The proposed approach of Knowledge distillation is based on model compression by training the smaller student model with the pre-trained teacher network. This method of implementing pre-trained teacher model to distil its knowledge to student model known as offline distillation. Our proposed approach outperforms all the methods by exhibiting the accuracy of 98.7% .

Table 1 depicts the comparison of the proposed methodology with existing methodologies for sarcasm detection.

Methodologies	Dataset Used	Result
Pattern-based Approach [29]	Twitter dataset	Precision– 91.1% Accuracy– 83.1%
Statistical Approach (SVM & Voting Classifier)	Twitter dataset	SVM Accuracy – 74.59% Voting classifier Accuracy– 83.53%
Transformer based approach (Recurrent CNN RoBERTa) [30]	SemEval Reddit Politics	SemEval Accuracy–82.0% Reddit Politics Accuracy – 79%
Multi-Rule based Ensemble feature selection model	Twitter API	Accuracy-95%
LSTM-CNN	Processed Tweets	Accuracy- 93 – 95%
Bi-directional LSTM using Multi-head Attention. Bi-LSTM	SARC dataset	Balanced, F-score – 77.48% Imbalanced, F-score – 56.79%
Bi-directional Long Short- Term Memory using Soft Attention function followed by Convolution network model. Bi-LSTM ConvNet	Balanced Dataset of SemEval & Imbalanced Dataset of Random Tweets	SemEval Dataset Accuracy– 97.87% Random Tweets Dataset Accuracy – 93.71%
DistilBERT (Proposed Method)	News Headline	Accuracy- 98%

Table 1. Result Analysis

Chapter 6

CONCLUSION & FUTURE SCOPE

In this report, we looked at machine learning, contextual learning, and deep learning approaches to detect sarcasm. Sarcasm Detection is one of the vast problems of NLP. Methodologies discussed so far are based on machine learning and deep learning algorithm which takes a lot of computation time and requires a robust system to handle the large datasets. Our proposed approach resolves this problem. Implementing the concept of Knowledge distillation to complex deep learning algorithms converges the complexity of the neural networks by maintaining efficiency and accuracy. This research is based on implementing the Knowledge distillation on the BERT model of deep learning on news headline datasets. Other deep learning models like Bi-LSTM with a soft attention mechanism followed by CNN also perform better with SemEval datasets with an accuracy of 93%. The proposed method of DistilBERT outperforms all other existing methodologies for sarcasm detection with an accuracy of 98.7%. In future, due to increase in the social media users it is very difficult to handle large amount of data with existing DL models. So our proposed approach is a best alternative to achieve efficient result on large datasets.

REFERENCES

- [1] Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., "Attention is all you need", Proc. NIPS, 2017.
- [2] Kumar, V. T. Narapareddy, V. Aditya Srikanth, A. Malapati and L.B. M. Neti, "Sarcasm Detection Using Multi-Head Attention Based Bidirectional LSTM," in *IEEE Access*, vol. 8, pp. 6388-6397, 2020, doi: 10.1109/ACCESS.2019.2963630.
- [3] Rajadesingan, A., Zafarani, R., & Liu, H. (2015, February). Sarcasm detection on twitter: A behavioral modeling approach. In *Proceedings of the eighth ACM international conference on web search and data mining* (pp. 97-106).
- [4] Bouazizi, M., & Ohtsuki, T. O. (2016). A pattern-based approach for sarcasm detection on twitter. *IEEE Access*, 4, 5477-5488.
- [5] Porwal, S., Ostwal, G., Phadtare, A., Pandey, M., & Marathe, M. V. (2018, June). Sarcasm detection using recurrent neural network. In *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 746-748). IEEE.
- [6] Muaad, A. Y., Jayappa Davanagere, H., Benifa, J. V., Alabrah, A., Naji Saif, M. A., Pushpa, D., ... Alfakih, T. M. (2022). Artificial intelligence- based approach for misogyny and sarcasm detection from Arabic texts. *Computational Intelligence and Neuroscience*, 2022.
- [7] Misra, R., Arora, P. (2019). Sarcasm detection using hybrid neural network. arXiv preprint arXiv:1908.07414.
- [8] Jain, D., Kumar, A., Garg, G. (2020). Sarcasm detection in mash-up language using soft-attention based bi-directional LSTM and feature-rich CNN. *Applied Soft Computing*, 91, 106198.
- [9] R. Kumar and A. Bhat, "An Analysis On Sarcasm Detection Over Twitter During COVID-19," 2021 2nd International Conference for Emerging Technology (INCET), 2021, pp. 1-6, doi: 10.1109/INCET51464.2021.9456392.

- [10] Isaac, Akileng Bhat, Aruna. (2022). A Conceptual Enhancement of LSTM Using Knowledge Distillation for Hate Speech Detection. 10.1007/978-981-16-4016-2 53.
- [11] Ndibanje, B., Kim, K. H., Kang, Y. J., Kim, H. H., Kim, T. Y., & Lee, H. J. (2019). Cross-method-based analysis and classification of malicious behavior by api calls extraction. *Applied Sciences*, 9(2), 239.
- [12] Webster, J. J., & Kit, C. (1992). Tokenization as the initial phase in NLP. In *COLING 1992 Volume 4: The 14th International Conference on Computational Linguistics*.
- [13] Jivani, A. G. (2011). A comparative study of stemming algorithms. *Int. J. Comp. Tech. Appl*, 2(6), 1930-1938.
- [14] Ingason, A. K., Helgadóttir, S., Loftsson, H., & Rögnvaldsson, E. (2008, August). A mixed method lemmatization algorithm using a hierarchy of linguistic identities (HOLI). In *International Conference on Natural Language Processing* (pp. 205-216). Springer, Berlin, Heidelberg.
- [15] Verma, P., Shukla, N., & Shukla, A. P. (2021, March). Techniques of sarcasm detection: A review. In *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)* (pp. 968-972). IEEE.
- [16] Dharwal, P., Choudhury, T., Mittal, R., & Kumar, P. (2017, December). Automatic sarcasm detection using feature selection. In *2017 3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)* (pp. 29-34). IEEE.
- [17] Kumar, H. K., & Harish, B. S. (2018). Sarcasm classification: a novel approach by using content based feature selection method. *Procedia computer science*, 143, 378-386.
- [18] Chaudhari, P., & Chandankhede, C. (2017, March). Literature survey of sarcasm detection. In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)* (pp. 2041-2046). IEEE.
- [19] Bharti, S. K., Naidu, R., & Babu, K. S. (2017, December). Hyperbolic feature-based sarcasm detection in tweets: a machine learning approach. In *2017 14th IEEE India Council International Conference (INDICON)* (pp. 1-6). IEEE.

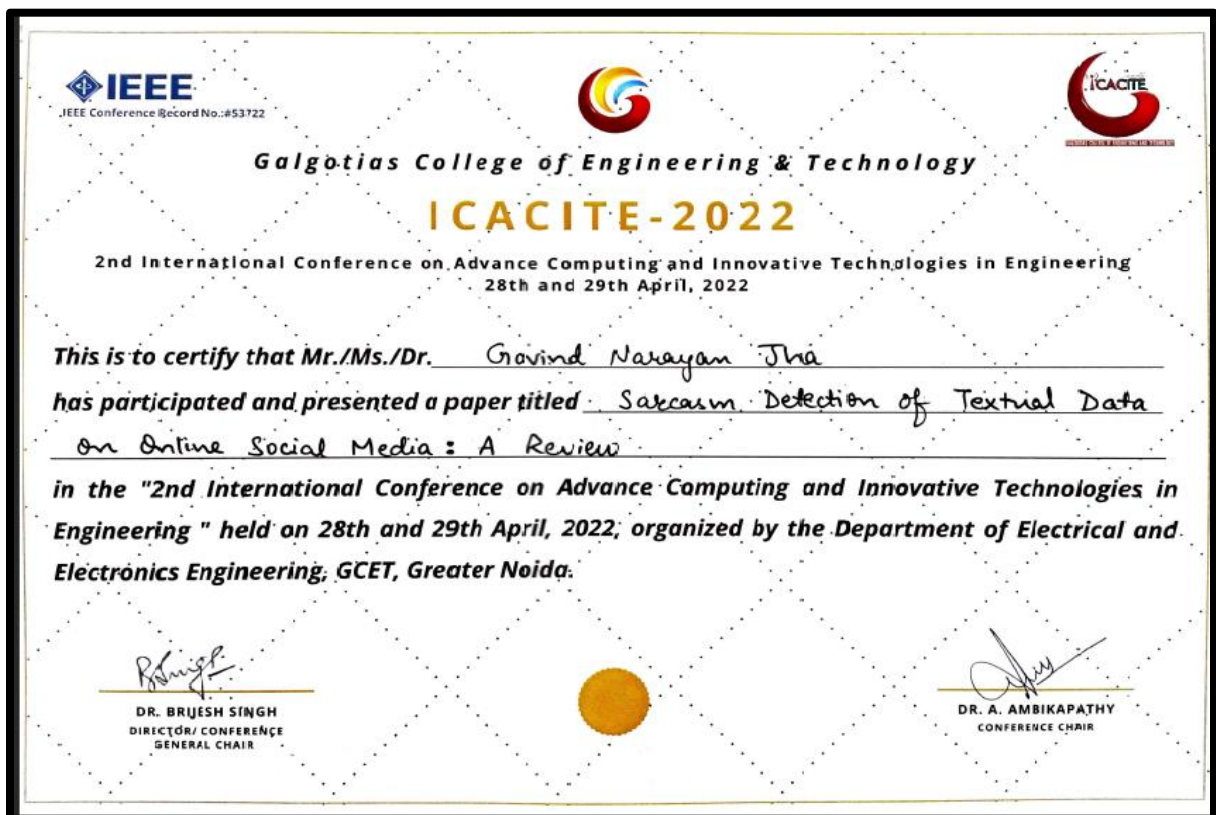
- [20] Shah, B., & Shah, M. (2021). A Survey on Machine Learning and Deep Learning Based Approaches for Sarcasm Identification in Social Media. In *Data Science and Intelligent Applications* (pp. 247-259). Springer, Singapore.
- [21] Onan, A., & Toçoğlu, M. A. (2021). A term weighted neural language model and stacked bidirectional LSTM based framework for sarcasm identification. *IEEE Access*, 9, 7701-7722.
- [22] Gupta, S., Shah, A., Shah, M., Syiemlieh, L., & Maurya, C. (2021, December). FiLMing Multimodal Sarcasm Detection with Attention. In *International Conference on Neural Information Processing* (pp. 178-186). Springer, Cham.
- [23] Hahn, S., Choi, H. (2019). Self-knowledge distillation in natural language processing. arXiv preprint arXiv:1908.01851.
- [24] <https://www.analyticsvidhya.com/blog/2022/01/knowledge-distillation-theory-and-end-to-end-case-study/>
- [25] Z. Chen, Le Zhang, Z. Cao and J. Guo, "Distilling the Knowledge From Handcrafted Features for Human Activity Recognition," in *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4334-4342, Oct. 2018, doi: 10.1109/TII.2018.2789925.
- [26] Yu, J., Liu, W., He, Y., Zhang, C. (2021). A mutually auxiliary multitask model with self-distillation for emotion-cause pair extraction. *IEEE Access*, 9, 26811-26821.
- [27] Savini, E., Caragea, C. (2022). Intermediate-Task Transfer Learning with BERT for Sarcasm Detection. *Mathematics*, 10(5), 844.
- [28] Nayak, D. K., & Bolla, B. K. (2022). Efficient Deep Learning Methods for Sarcasm Detection of News Headlines. In *Machine Learning and Autonomous Systems* (pp. 371-382). Springer, Singapore.
- [29] Bouazizi, M., & Ohtsuki, T. O. (2016). A pattern-based approach for sarcasm detection on twitter. *IEEE Access*, 4, 5477-5488.
- [30] Potamias, R. A., Siolas, G., Stafylopatis, A. G. (2020). A transformer- based approach to irony and sarcasm detection. *Neural Computing and Applications*, 32(23), 17309-17320

LIST OF PUBLICATIONS

1. Aruna Bhat and Govind Narayan Jha, *Sarcasm Detection of Textual Data on Online Social Media: A Review*, International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE 2022) Greater Noida, India, 28th-29th April 2022.

Indexed by Scopus and Google Scholar.

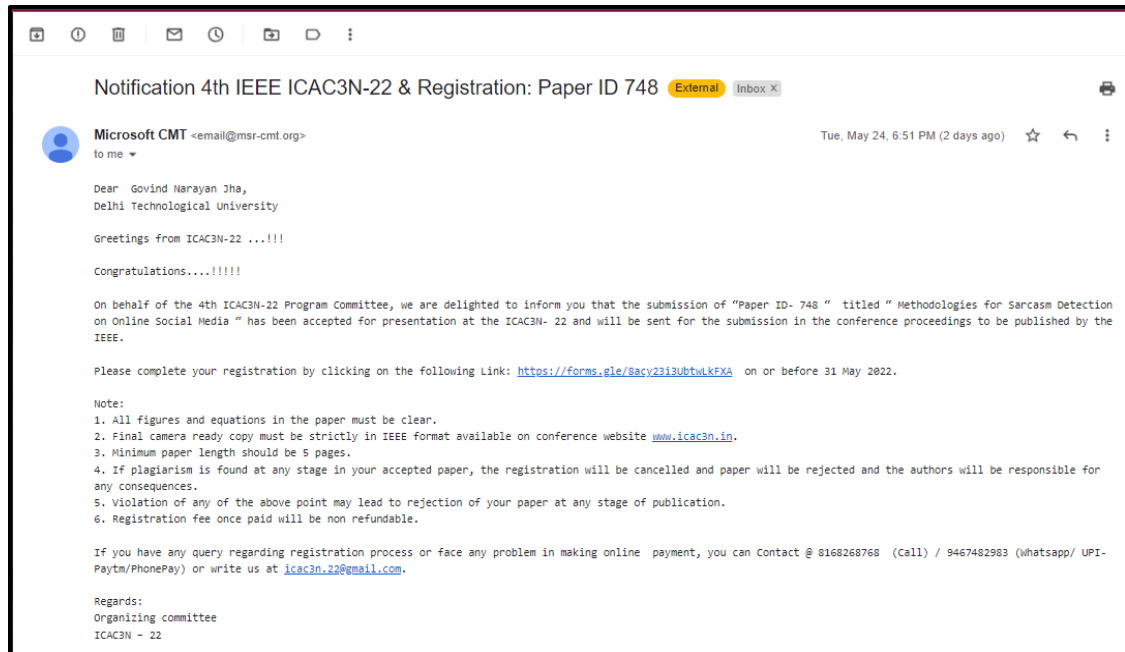
CERTIFICATE



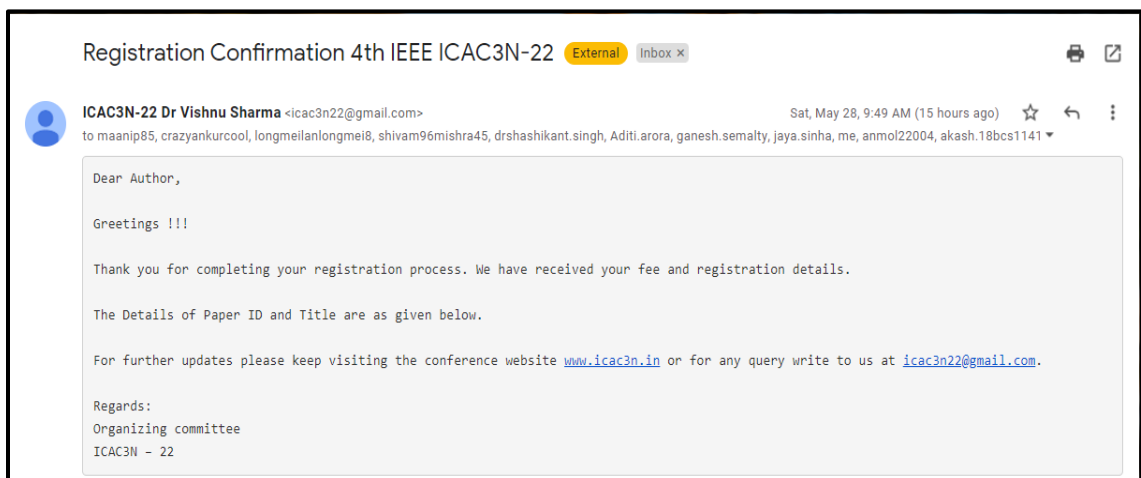
2. Govind Narayan Jha and Aruna Bhat, *Methodologies for Sarcasm Detection on Online Social Media*, 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N–2022), Greater Noida, India. 16th -17th December.

Indexed by Scopus and Google Scholar.

Proof of Acceptance



Registration Confirmation





Transfer Successful

Reference ID 214612580405
Mode IMPS
Paid to Account Galgotias College of Engineering and Technology
6420000100006852
Amount ₹ 5,000.00
From Account XX-XX-XX-XX-XX-42
On 26/05/2022 12:28:57
Remarks Research paper

Tip: Want to share this success page with beneficiary ? Use  icon on top right.



PAPER NAME

govind2K20CSE09.pdf

AUTHOR

RAJU KUMAR

WORD COUNT

7099 Words

CHARACTER COUNT

39200 Characters

PAGE COUNT

45 Pages

FILE SIZE

1.6MB

SUBMISSION DATE

May 29, 2022 12:14 PM GMT+5:30

REPORT DATE

May 29, 2022 12:16 PM GMT+5:30

● 16% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 8% Internet database
- 7% Publications database
- Crossref database
- Crossref Posted Content database
- 12% Submitted Works database

● Excluded from Similarity Report

- Bibliographic material