Sarcasm Detection in Tweets: A Feature-based Approach using Supervised Machine Learning Models

Arifur Rahaman[1], Ratnadip Kuri[2], Syful Islam[3]*, Md. Javed Hossain[4], Mohammed Humayun Kabir[5]

Dept. of Computer Science and Telecommunication Engineering
Noakhali Science and Technology University, Bangladesh[1,2,3,4,5]
Dept. of Computer Science and Engineering, Sonargaon University, Bangladesh[1]

Abstract—Sarcasm (i.e., the use of irony to mock or convey contempt) detection in tweets and other social media platforms is one of the problems facing the regulation and moderation of social media content. Sarcasm is difficult to detect, even for humans, due to the deliberate ambiguity in using words. Existing approaches to automatic sarcasm detection primarily rely on lexical and linguistic cues. However, these approaches have produced little or no significant improvement in terms of the accuracy of sentiment. We propose implementing a robust and efficient system to detect sarcasm to improve accuracy for sentiment analysis. In this study, four sets of features include various types of sarcasm commonly used in social media. These feature sets are used to classify tweets into sarcastic and non-sarcastic. This study reveals a sarcastic feature set with an effective supervised machine learning model, leading to better accuracy. Results show that Decision Tree (91.84%) and Random Forest (91.90%) outperform in terms of accuracy compared to other supervised machine learning algorithms for the right features selection. The paper has highlighted the suitable supervised machine learning models along with its appropriate feature set for detecting sarcasm in tweets.

Keywords—Machine learning; detection; sarcasm; sentiment; tweets

I. INTRODUCTION

Sarcasm detection in opinion mining is an essential tool with various applications, including health, security, and sales [1, 2]. Several organizations and companies have shown their interest in studying tweets data to know people's opinion towards popular products, political events or movies. Millions of tweets are posted every day, which increase the content of twitter tremendously.

However, microblogging social media (i.e., maximum 140 characters in every tweet) and containing informal language essentially makes it quite tricky to understand users' sentiment and perform sentiment analysis. Additionally, sarcasm poses a challenge in sentiment analysis and causes the misclassification of people's opinions. Hence, it leads to reduced accuracy of sentiment analysis. People use sarcasm to mock or convey contempt through a sentence or while speaking. People apply positive words to reveal gloomy feelings. In recent days, sarcasm or irony is very common in social media, although it is challenging to detect.

The cutting-edge approaches of opinion mining and sentiment analysis are prone to unsatisfactory performances while analyzing social media data. Maynard and others [3] proposed that detecting sarcasm during sentiment analysis might significantly improve performance. Consequently, the necessity for an effective method to identify sarcasm arises.

In this paper, we propose an effective method to identify a sarcastic tweet. Our strategy considers the various types of sarcasm indicating features such as Lexical-based Features, Sarcastic-based Features, Contrast-based Features, Context-based Features, and detects the sarcastic tweets using multiple supervised machine learning models based on extracted features. We suggest an effective machine learning model and feature set to better perform sarcasm detection in sentiment analysis and get better accuracy, which is explained at the end of the evaluation and the result analysis parts of this paper.

The main contributions are as follows:

1) To detect sarcasm, we use a set of machine learning classification algorithms along with a variety of features to identify the best classifier model with significant features, which leads to recognize of the sarcasm in tweets to get better performance of sentiment analysis.

2) We propose the right set of features that lead to better accuracy, which is presented in the result analysis part of this paper.

3) Analysis results present that Decision Tree (91.84%) and Random Forest (91.90%) outperform the accuracy compared to Logistic Regression, Gaussian Naive Bayes, and Support Vector Machine for the different features set up.

The remainder of this paper is arranged as follows: Section 2 explains the literature review, and Section 3 demonstrates the sarcasm recognition process. Section 4 illustrates our findings, and Section 5 concludes this work.

II. LITERATURE REVIEW

Many research articles and publications motivated us to work with this topic; a few of them are discussed here in detail. The authors, Sana Parveen, Sachin N. Deshmukh [4], suggested a methodology to recognize the sarcasm on Twitter using Maximum Entropy and Support Vector Machine (SVM). Firstly, they created two datasets from collected

Twitter data. One dataset has sarcastic tweets and another dataset without sarcastic tweets in training data. Penn treebank is used to tag POS with each word. Maximum Entropy and SVM are used to classify tweets after features extraction related to sentiment, syntactic, punctuation, and pattern. Finally, they got more accuracy for Maximum Entropy than SVM. The authors, Anukarsh G Prasad, Sanjana S, Skanda M Bhat, BS Harish [5], suggested a technique based on the slang and emojis used in tweets to identify sarcastic and non-sarcastic tweets. They took into consideration slang and emoji values according to the slang and the emoji dictionary. Afterward, these values are used to classify sarcasm, applying Random Forest, Gaussian Naive Bayes, Gradient Boosting, Logistic Regression, Adaptive Boost, Logistic Regression, and Decision Tree. They suggested an effective classifier using slang and emoji dictionary mapping to produce the most satisfactory performance. A new technique was suggested by Sreelakshmi K, Rafeeque P C [6] using context incongruity to define sarcasm on Twitter. To recognize the irony, they considered various features such as linguistic, sentiment, and context features. Both the sarcastic and non-sarcastic tweets are collected and pre-processed. They used Simple Vector Machine (SVM) and Decision Tree as a sarcasm classifier and produced a satisfactory level. A significant study from the linguistic sector tells us that lexical factors such as adjectives and adverbs, interjections, and punctuations play a considerable role in sarcasm detection [7]. In sentiment analysis, many researchers used machine learning methods such as Maximum Entropy, Naïve Bayes, and Support Vector machine because these algorithms tend to outperform the other algorithms in text classification [8,9]. Buscaldi and others (2012) [10] addressed the features which lead to the sarcasm classification. It also provides an in-depth description of how the different features contribute to the classification. Barbieri F. And Saggion H.'s work (2014) [11] dealt with the automatic detection of sarcasm on Twitter data. They divided the results of the classification of tweets into sarcastic or non-sarcastic classes. They do it depending on frequency (the gap between rare and common words), written-spoken (written-spoken style uses), intensity (intensity of adverbs and adjectives). And also, analyze structure (length, punctuation, emoticons, links), sentiments (the gap between positive and negative terms), synonyms (common vs. rare synonyms use), and Ambiguity (a measure of possible ambiguities). Based on the features mentioned above, they proposed a classification algorithm and claimed 71% accuracy on irony detection. The authors, David B. and Noah A. S. (2016) [12], improved the classification method by adding the history of tweets and author profiles, which helps in the classification process. The article presents accuracy for different contexts ranging from 70% and upwards. Parmar and others (2018) [13] pointed out a few of the present challenges to classify sarcastic tweets, such as the nature of the collected tweet, the presence of uncommon words, abbreviations, and slang, that are more informal and no predefined structure for sarcasm identification in Twitter. Ren, Y., Ji, D., and Ren, H. suggested two distinct context-augmented neural models for detecting sarcasm in the text [14]. Prasad and others [15] examined numerous classification approaches in which they noticed that Gradient Boost (GB) showed the best performance. Another study stated a novel method for identifying sarcasm in tweets by combining two classification techniques, Support Vector Machine with N-gram features [16]. Karthik Sundararajan and others proposed a sarcasm detecting approach and irony type using Multi-Rule Based Ensemble Feature Selection Model [17]. Siti Khotija, Khadijah Tirtawangsa, Arie A Suryani suggested a context-based method to detect sarcasm in tweets based on Long Short-Term Memory (LSTM) [18]. According to another paper, classifier's performance is vital in sarcasm predictions in opinion mining [19].

Moreover, the nature of classifiers can also play an essential role in sarcasm detection. However, in tweets, limited studies have approached the efficiency of sarcasm detection methods with valuable features. Hence, this study investigates the principal sarcasm classifiers, classification performance, and the selection of features that dominate such performance.

Besides, the paper drives motivation from numerous works, as mentioned earlier, and intends to produce better performance in sarcasm detection.

III. Proposed Methodology

The structure of suggested methodology is depicted in Fig. 1 and mainly consists of three modules such as tweet pre-processing, feature engineering, and sarcasm recognition modules.

A. Tweets Dataset

Twitter's streaming API was used to collect tweets. To obtain sarcastic tweets, we request the API for tweets having the hashtag "#sarcasm". Similarly, for no sarcastic tweets, we collected tweets regarding different topics and eliminated ones that include any hashtag indicating sarcasm. We collect a total of 76799 tweets having two categories, sarcastic (37583) and non-sarcastic (39216) tweets. 0(zero) is used to indicate non-sarcastic tweets, while 1(one) for sarcastic tweets. The dataset contains two columns as Label and Tweet. The Label column has 1 or 0 to present, whether it is sarcastic or not while the Tweet column has tweets.
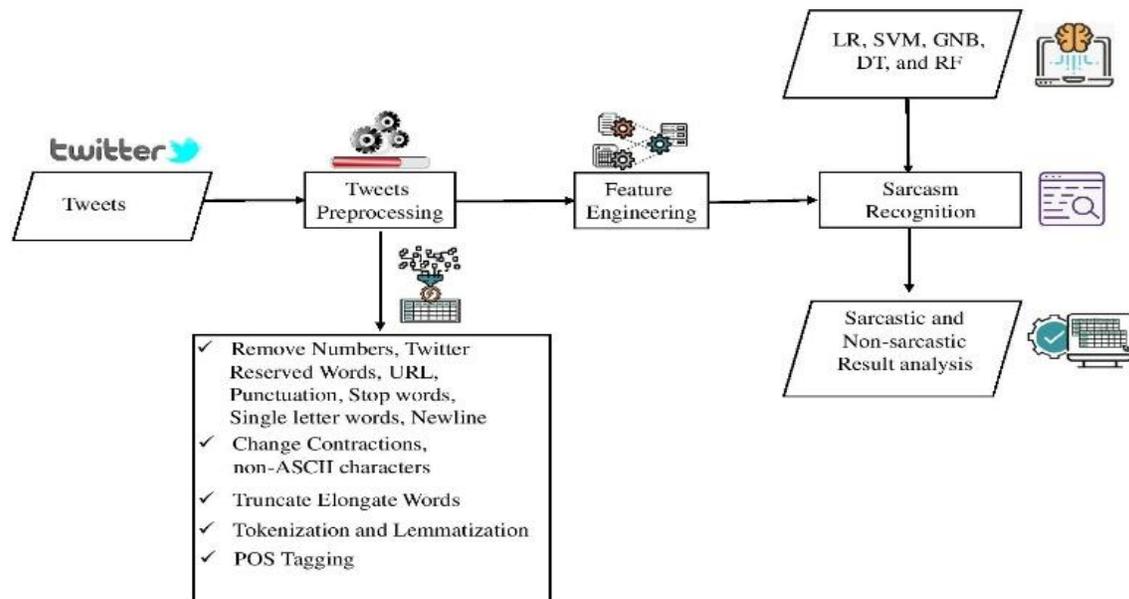
Fig. 1.   Proposed Methodology.

B. Tweets Preprocessing

1) Noise removal: We have removed numbers, newlines, non-ASCII characters, twitter preserved words, a single word tweet and some common string literals to speed up the feature mining process.

2) URLs removal: URLs (Uniform Resource Locators) in the tweets are references to a location on the web but do not provide any additional information. They are removed in the pre-processing phase of the sarcasm detection process.

3) Stop words removal: One of the significant forms of pre-processing is to filter out useless data referred to as stop words. Stop words include mainly articles (a, an, the), prepositions (in, of, to, and so on), along with other very commonly used words. They, indeed, don't have any contribution to detect sarcasm in the sentence. Therefore, they are removed before further processing of data. For example, after removing common words from "I don't know how to swim", we have "know and swim" words remaining.

4) Truncated elongated word: Tweets contain repeated characters in a word such as gooooood, loooove, moooove and many more styles. These words usually indicate sarcasm in tweets. We did not apply this preprocess step while counting the number of repeated letter segments and vowel repetitions in the tweet. However, we get the base word for these words, for instance, good, love, move, respectively, to extract other features.

5) Punctuation removal: Punctuations such as "#$%&'()*+,-./:;<=>@[\]^_`{|}~" are discarded before extracting features from tweets, though some special punctuations, for example, ?, !, … are preserved which sometimes indicate sign of sarcasm.

6) Replace contraction and acronym: Contractions are n't, aren't, wasn't, can't, couldn't, haven't, won't, shan't, shouldn't and many more. We replaced all of them with their full form so that we can analyze a single word. Moreover, we use forms of common acronyms used in tweets such as hella, lhh, lmao, jk and so on.

7) Normalization: To normalize tweets, we apply tokenization and lemmatization steps. Tokenization helps to create words list from tweets while lemmatization finds the base form of any provided word, for instance, made to make and loved to love. The base form of a word, in most cases, supports identifying tweet sentiment.

8) POS Tagging: It is the method of matching a word to its morphological class, which supports learning its role inside the sentence. Necessary parts of speech counted in POS Tagging are Noun, Verb, Adverbs, and Adjectives. Part-of-speech taggers essentially take a series of words as input and produce a list of tuples as output, where every word is connected with the relevant tag.

C. Feature Engineering

Being a modern form of communication, sarcasm is used for various purposes that fall in prevailing in three categories:

1) Irony as wit: when used as a wit, sarcasm is applied to be funny; the person applies some particular sorts of speeches, favors to exaggerate, or adopts a distinct tone from that when he speaks usually. In social networks, voice tones are transformed into particular kinds of writing: use of uppercase letter words, ellipsis, letter repetition, quote repetition, question marks, interjections, and exclamation, as well as some sarcasm-related emoticons.

2) Irony as whimper: when used as a whimper, sarcasm is used to reveal how annoyed or irritated the person is. Consequently, it stimulates to explain how wicked the circumstance is, using exaggeration and highly positive feelings to express a negative state.

3) Irony as an escape: it attributes to the circumstance when the person wishes to pretend to give a precise answer. Hence, it makes the presence of sarcasm. In this case, the person applies perplexing sentences, unusual words, some common expressions, and slangs.

We extracted four sets of features based on the assumption mentioned above: Lexical, Sarcastic, Contrast, and Context-based.

4) Lexical-based features: We extract seven lexical or textual-related features. They are Noun, verb, adverb, adjective having a higher impact in a tweet than any other parts of speech. We count them individually according to individual tweets. Moreover, intensifiers such as absolutely, amazingly, awfully, ridiculously, bloody, excessively, outrageously, strikingly, tremendously and so on, are sometimes used to show exaggeration in the tweet to express negative feelings through positive intensifiers and vice versa. Therefore, we also count the number of positive and negative intensifiers present in every tweet. Lastly, the whole tweet's sentiment is calculated, which reveals the overall polarity of the tweet.

5) Sarcastic-based features: Generally, people tend to make complicated sentences or use rare words to make them vague to the listener or reader to get a definite answer in other events. Indeed, when people use sarcasm as avoidance, they intend to hide their actual feeling or sentiment by hiding them in fun. Therefore, we derive the following features. People sometimes try to convey the irony message through punctuations, such as exclamation, question marks, and repeated ellipsis. Hence, we count the number of exclamations, question marks and repeated ellipsis. Additionally, interjections are 'ha-ha', 'ho-ho', 'ho-ho-ho', 'oh', 'ouch', 'ow', 'shh', 'super' 'kidding', 'ah', 'aha', 'aww', 'nah', 'yay', 'uh', 'bah', 'bingo', 'boo', 'bravo', 'brilliant' and many more. People use them to express their feeling in different ways, which help to identify sarcastic tweet. Words like loooove, gooooood, moooove having repeated letters more than three or equals to three also probable indications of mockery tweets and repetition of vowels also denote the same thing. So, we count the number of repeated letter segments and vowel repetition in the tweet. It is worthy to note here, we extract this feature before removing repeated letters. On the other hand, people apply the uppercase word, for instance, GOOD, AWESOME and part of the word as uppercase, for example, gOOd, aWESome, to reveal their irony feeling. We find their number as well. We compute the number of laughter namely 'lol', "lhh", "jk", 'wow', 'kidding', 'ha ha', 'ha-ha', 'haha', 'rofl', 'roflmao', 'lmao', 'wtf'. Emoji are the facial emotions such as laughter created by typing a series of keyboard symbols, which are normally used to convey the author's attitude, feeling, or intended tone. In particular, sarcastic Emoji are ones sometimes used with sarcastic or ironical statements (e.g., ":P"). We consider not only rare sarcastic words but also very common words used in sarcastic tweets.

Sarcastic slangs as examples of 'ayfkmwts', 'fubar', 'kyso', 'lhh', 'lmao', 'stfw', 'stf',' roflmao', 'rofl' are used in the tweet, and we take into consideration them too. We calculate the number of repeated quotes in the tweet as letter repetition. Besides, hashtags also contain emotional content.

In many cases, they are employed to ambiguate the tweet's real purpose carried in the message. Accordingly, we also calculate the sentiment score of hashtags. Finally, we calculate polarity summation (+1 for positive whereas -1 for negative polarity) for n-grams such as bigrams and trigrams.

6) Contrast-based features: We then outline four features that interpret whether there is a contrast between the various elements. By comparison, we indicate a negative element's coexistence and a positive one within the identical tweet. We calculate emoji polarity flip, polarity flip of sentiment, the number of positive and negative words. Before counting positive and negative words, we convert the following word of a negation word (e.g., "not", "never", etc.) into antonym, then find the number of positive words and that of negative words.

7) Context-based features: For context-related features, we find the number of users mentions and hashtags included in the tweet. People tend to express their feelings by applying various types of hashtags and sometimes use user mention as well.

D. Sarcasm Recognition

Usually, supervised machine learning models and lexicon-based approaches are used in opinion mining and text classification. The former includes various techniques such as K-Nearest Neighbor, Support Vector Machine (SVM), Linear Regression, Logistic Regression (LR) [20], Gaussian Naive Bayes (GNB)[1], Decision Tree (DT), Random Forest (RF), Neural Networks, Linear Discriminant Analysis (LDA) and so on. The latter has two strategies: a dictionary-based approach and a corpus-based approach.

However, for sarcasm detection in our proposed approach, we use five machine learning classifiers, such as Support Vector Machine, Logistic Regression, Gaussian Naive Bayes, Decision Tree and Random Forest, to examine which one performs better with our extracted features.

We commonly need two sets of data in machine learning-based classification, such as training data and test data sets. The classifier uses training data set to learn from the data and build the model, while the test data set is applied to validate the classifier's performance. In our case, we use the 10-fold cross-validation technique to split the dataset into training and test sets. Cross-validation is a method to evaluate predictive models by splitting the dataset into a training data set for training the model and a test data set to evaluate it.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Evaluation

We have evaluated the effective classifier and feature set depending on three metrics: precision, recall, and F1-score.

---

[1] https://iq.opengenus.org/gaussian-naive-bayes/

Precision: Precision presents the ratio between the True Positives and all actual Positives in the dataset. Additionally, precision indicates the number of Positive class predictions which belong to the Positive class. It denotes TP (True Positive) divided by the sum of TP (True Positive) and (FP) False Positive.

Recall: Recall presents the percentage of correctly identifying the True Positive class. It denotes TP (True Positive) divided by the sum of TP (True Positive) and (FN) False Negative.

F1-score: F1-score presents the weighted or Harmonic average of Precision and Recall. Hence, it takes into consideration both False Positive and False Negatives. The formula denotes it: (2*Precision*Recall) / (Precision + Recall).

1) Evaluating effective sarcasm classifier: We have evaluated five machine learning classifiers depending on precision, recall and F1-score to investigate which outperforms. According to Table I, Random Forest shows the highest precision, and Naïve Bayes presents the lowest value for precision. Regarding Recall and F1 metrics, the Decision Tree classifier performs better than other classifiers, 91.72% and 91.67%, respectively. Overall, the Decision tree outperforms different classifiers.

2) Evaluating of effective feature set: We have evaluated four feature sets with three popular classifiers, such as Support Vector Machine, Decision Tree and Random Forest. Here, also we have investigated three metrics: precision, recall and F1-score. According to our evaluation presented in Table II, the Sarcastic feature set produces consistently better values for precision, recall and F1 than other feature sets. In contrast, the context-based feature set shows the lowest values in all cases.

B. Results

This section is explained in three separate subsections as follows:

- Investigation of the accuracy metric depending on various features.

- Explanation of our observation for various feature combinations.

- Analysis of the variation in classification accuracy for adding categories incrementally.

We have used accuracy metric to perform results analysis.

Accuracy*:* Accuracy indicates the ratio between the total number of accurate predictions and the total number of possible predictions in test data. It is calculated as the sum of TP (True Positive) and TN (True Negative) divided by the sum of total Positives and Negative classes.

1) Performances of each set of features: According to the given Table III and Fig. 2 above, it is clear that sarcastic features in our study have a much better contribution in sarcasm detection in tweets than any other features such as lexical, contrast, and context features. For the rest of the

feature sets, lexical features have better accuracy than contrast-based features, while context-based features have far less impact on sarcasm detection. Decision Tree performs consistently better in terms of classification technique, and it reaches around 80 percent accuracy for sarcastic features merely. Logistic Regression shows the lowest accuracy for lexical and sarcastic features, while contrast and context features are less useful for the Random Forest technique. As sarcastic features produce far more accuracy, it is worthy to note that the proper selection of sarcastic feature set can increase the accuracy instead of selecting any other features.

TABLE I. EVALUATION OF EFFECTIVE SARCASM DETECTION CLASSIFIER. RESULT SHOWS THAT THE DECISION TREE OUTPERFORMS OTHER CLASSIFIERS

Classifiers	Precision	Recall	F1-score
LR	72.12	42.43	53.43
SVM	90.06	85.93	85.36
GNB	63.43	84.18	72.35
DT	91.62	**91.72**	**91.67**
RF	**91.9**	91.5	90.33

TABLE II. EVALUATION OF EFFECTIVE FEATURE SET ALONG WITH VARIOUS CLASSIFIERS. RESULT SHOWS THAT THE SARCASTIC FEATURE SET PRODUCES CONSISTENTLY BETTER VALUES FOR PRECISION, RECALL AND F1 THAN OTHER FEATURE SETS

Classifiers	Features	Precision	Recall	F1-score
SVM	Lexical	58.85	62.79	60.75
	Sarcastic	**77.83**	**68.05**	**72.61**
	Contrast	60.99	35.68	45.02
	Context	0	0	0
DT	Lexical	64.68	60.77	62.67
	Sarcastic	**82.57**	**77.72**	**80.07**
	Contrast	57.71	74.76	65.14
	Context	55.59	37.5	44.79
RF	Lexical	56.69	53.31	54.95
	Sarcastic	**80.19**	**74.98**	**77.5**
	Contrast	39.04	41.73	40.34
	Context	21.93	20.36	21.11

TABLE III. ACCURACY (%) ON EVERY FEATURE SET. OUR ANALYSIS SHOWS THAT SARCASTIC FEATURES ARE USEFUL TO OBTAIN HIGHER CLASSIFICATION ACCURACY THAN OTHERS

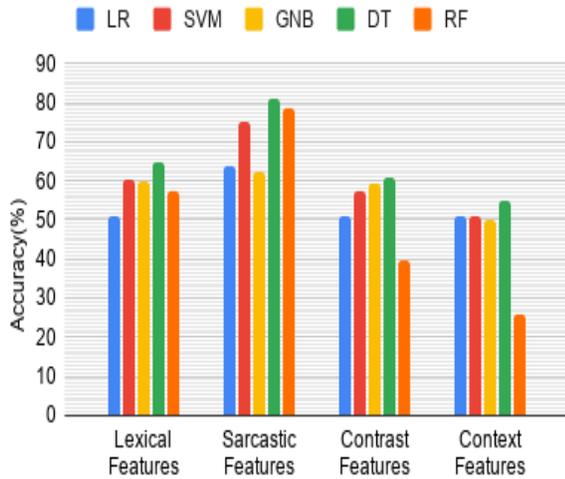| Classifier	Lexical Features	Sarcastic Features	Contrast Features	Context Features
LR	51.10	**63.83**	51.10	51.10
SVM	60.33	**74.90**	57.39	51.10
GNB	59.83	**62.19**	59.56	49.96
DT	64.59	**81.08**	60.87	54.79
RF	57.25	**78.71**	39.65	25.61

Fig. 2.   Graphical view of Accuracy (%) on Every Feature Category. We Observe that Sarcastic Features are Dominating to get Higher Sarcasm Detection Accuracy.

2) Accuracy for various feature combinations: As of Table IV and Fig. 3 mentioned below, sarcastic (S) and contrast (C) based features together show the highest accuracy for DT and RF than the other feature combinations. The analysis result shows that lexical (L) and sarcastic feature combinations achieve more accuracy than sarcastic and context features. Mainly, GNB produces the highest accuracy for lexical and sarcastic features set. For all classification techniques, contrast and context (Cx) features together represent the lowest accuracy. As explained, except for lexical and sarcastic feature sets, it is found that the sarcastic and contrast features combinedly are the minimal feature set which leads to getting higher accuracy than the remaining combination sets. Although the Decision Tree classification technique has consistently higher accuracy than random forest and others, it exhibits lower accuracy for lexical and sarcastic feature combinations. Naïve Bayes and Logistic Regression are stabilized with almost 63% accuracy, while accuracy for SVM fluctuates between around 58% and 87%. Overall, the Decision Tree produces maximum accuracy, nearly 91% for the minimal sarcastic and context-based feature set.

TABLE IV.   ACCURACY (%) FOR COMBINED FEATURE SETS. SARCASTIC (S) AND CONTRAST (C) BASED FEATURES TOGETHER SHOW THE HIGHEST ACCURACY FOR DT AND RF THAN THE OTHER FEATURE COMBINATIONS

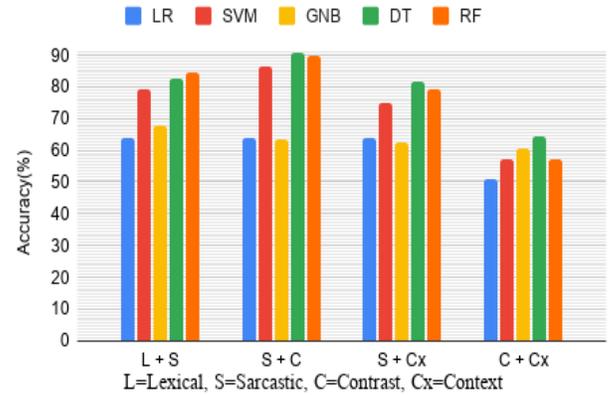| Classifier	L + S	S + C	S + Cx	C + Cx
LR	63.83	63.83	63.83	51.10
SVM	79.34	86.59	74.90	57.39
GNB	67.95	63.44	62.63	60.32
DT	82.79	**90.99**	**81.86**	**64.54**
RF	**84.66**	89.81	79.38	57.21

Fig. 3.   Accuracy for Various Category Combinations. Sarcastic (S) and Contrast (C) based Features Produce the Highest Accuracy for DT and RF than the other Feature Combinations.

3) Accuracy for incrementally added category: Depending on our findings as of Table V and Fig. 4, we observe that the sarcastic-based features dominated the accuracy metric. We fixed them as a base feature set and added feature sets incrementally to understand how accuracy changed with more feature sets. The addition of two (sarcastic, context) and three (sarcastic, contrast and lexical) feature sets show almost similar accuracy. Therefore, it is observable that lexical features with sarcastic and contrast features have far less impact on accuracy. It is not necessary to add more features to get better accuracy at all. Analysis results reveal that Decision Tree (91.84%) and Random Forest (91.90%) exceed the accuracy compared to Logistic Regression, Gaussian Naive Bayes, and Support Vector Machine for the different features selection.

C. Discussion

Looking at the values obtained in our sarcasm recognition experiments, it seems that the sarcastic feature set has more contribution to sarcasm detection in tweets among the four feature sets. In fact, this feature set's accuracy is constantly more excellent than other feature sets with all classifiers, and the Decision Tree shows maximum accuracy (approximately 81%) for the sarcastic feature set.

TABLE V.   ACCURACY (%) FOR THE INCREMENTALLY ADDED FEATURE SETS. WE FOUND THAT THE SARCASTIC BASED FEATURES DOMINATED THE ACCURACY METRIC, AND ANALYSIS RESULTS REVEAL THAT DECISION TREE (91.84%) AND RANDOM FOREST (91.90%) PRESENT HIGHER PERFORMANCE

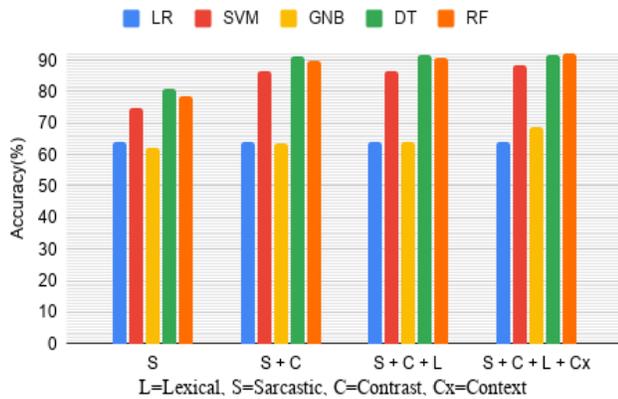| Classifier	S	S + C	S + C + L	S + C + L + Cx
LR	63.83	63.83	63.83	63.83
SVM	74.90	86.59	86.59	88.48
GNB	62.19	63.44	64.16	68.53
DT	**81.07**	**91.04**	**91.51**	91.84
RF	78.71	89.81	90.60	**91.90**

Fig. 4. Accuracy for the Incrementally Added Category. Decision Tree and Random Forest Classifiers show Better Accuracy along with the Sarcastic Feature Set.

Now turning to the experiment on different feature set combinations. We analyzed four distinct combinations to see which combination outperforms others. We combined the sarcastic feature set with lexical, contrast and context-based feature-set separately and contrast and context together for this experiment. We found higher accuracy (around 90%) for the sarcastic and contrast-based feature set. It appears that adding more features does not increase the model performance and feature selection is the central part of the efficient classification.

Lastly, we performed tests to observe which classifier outperforms if we increase the features, considering the sarcastic feature set as the base feature set. According to our observation, all feature sets' accuracy is not far more significant than other feature combinations. Hence, it is not always practical to add more feature sets to get better accuracy. However, the Decision Tree presents overall higher accuracy in all cases.

V. CONCLUSION AND FUTURE WORK

In this paper, we have suggested an improved model for detecting sarcasm in sentiment analysis. According to results, sarcastic features are more dominating than other features in sarcasm detection in tweets. Results show higher accuracy with sarcastic-based features for all classifiers we have used in our study. Moreover, the Decision Tree presents the highest accuracy (around 81%) with sarcastic-based features. Combining contrast-based features with sarcastic features increases the accuracy at approximately 90% for the Decision Tree. Therefore, it seems that it is not enough to add more features to obtain high accuracy. Though, the selection of the suitable feature set is the central part of the effective classification. Finally, we have evaluated all classifiers with incrementally added features, and findings reveal overall higher accuracy for the Decision Tree. We will study how to apply our proposed approach to improve sentiment analysis and opinion mining performances in future work. Additionally, we are also interested in context and pattern-based sarcasm detection in sentiment analysis.

REFERENCES

[1] S. Jain, V. Hsu, "The lowest form of wit: Identifying sarcasm in social media," 2015.

[2] S. Saha, J. Yadav, P. Ranjan, "Proposed approach for sarcasm detection in twitter," Indian Journal of Science and Technology, 10(25), 1–8, 2017.

[3] D. G. Maynard, M. A. Greenwood, "Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis," in Lrec 2014 proceedings, ELRA, 2014.

[4] S. Parveen, S. N. Deshmukh, "Opinion Mining in Twitter–Sarcasm Detection," Politics, 1200, 125, 2017.

[5] A. G. Prasad, S. Sanjana, S. M. Bhat, B. Harish, "Sentiment analysis for sarcasm detection on streaming short text data," in 2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA), 1–5, IEEE, 2017.

[6] K. Sreelakshmi, P. Rafeeque, "An effective approach for detection of sarcasm in tweets," in 2018 International CET Conference on Control, Communication, and Computing (IC4), 377–382, IEEE, 2018.

[7] R. Kreuz, G. Caucci, "Lexical influences on the perception of sarcasm," in Proceedings of the Workshop on computational approaches to Figurative Language, 1–4, 2007.

[8] A. R. Naradhipa, A. Purwarianti, "Sentiment classification for Indonesian message in social media," in 2012 International Conference on Cloud Computing and Social Networking (ICCCSN), 1–5, IEEE, 2012.

[9] B. Pang, L. Lee, S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," arXiv preprint cs/0205070, 2002.

[10] A. Reyes, P. Rosso, D. Buscaldi, "From humor recognition to irony detection: The figurative language of social media," Data & Knowledge Engineering, 74, 1–12, 2012.

[11] F. Barbieri, H. Saggion, "Automatic Detection of Irony and Humour in Twitter." in ICCC, 155–162, 2014.

[12] D. Bamman, N. Smith, "Contextualized sarcasm detection on twitter," in Proceedings of the International AAAI Conference on Web and Social Media, volume 9, 2015.

[13] K. Parmar, N. Limbasiya, M. Dhamecha, "Feature based composite approach for sarcasm detection using MapReduce," in 2018 second international conference on computing methodologies and communication (ICCMC), 587–591, IEEE, 2018.

[14] Y. Ren, D. Ji, H. Ren, "Context-augmented convolutional neural networks for twitter sarcasm detection," Neurocomputing, 308, 1–7, 2018.

[15] A. G. Prasad, S. Sanjana, S. M. Bhat, B. Harish, "Sentiment analysis for sarcasm detection on streaming short text data," in 2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA), 1–5, IEEE, 2017.

[16] P. Tungthamthiti, K. Shirai, M. Mohd, "Recognition of sarcasm in microblogging based on sentiment analysis and coherence identification," Information and Media Technologies, 12, 80–102, 2017.

[17] K. Sundararajan, A. Palanisamy, "Multi-rule based ensemble feature selection model for sarcasm type detection in twitter," Computational intelligence and neuroscience, 2020.

[18] S. Khotijah, J. Tirtawangsa, A. A. Suryani, "Using lstm for context based approach of sarcasm detection in twitter," in Proceedings of the 11th International Conference on Advances in Information Technology, 1–7, 2020.

[19] S. M. Sarsam, "Reinforcing the decision-making process in chemometrics: Feature selection and algorithm optimization," in Proceedings of the 2019 8th international conference on software and computer applications, 11–16, 2019.

[20] Hilbe, Joseph M., Logistic Regression Models. CRC Press, 2009.