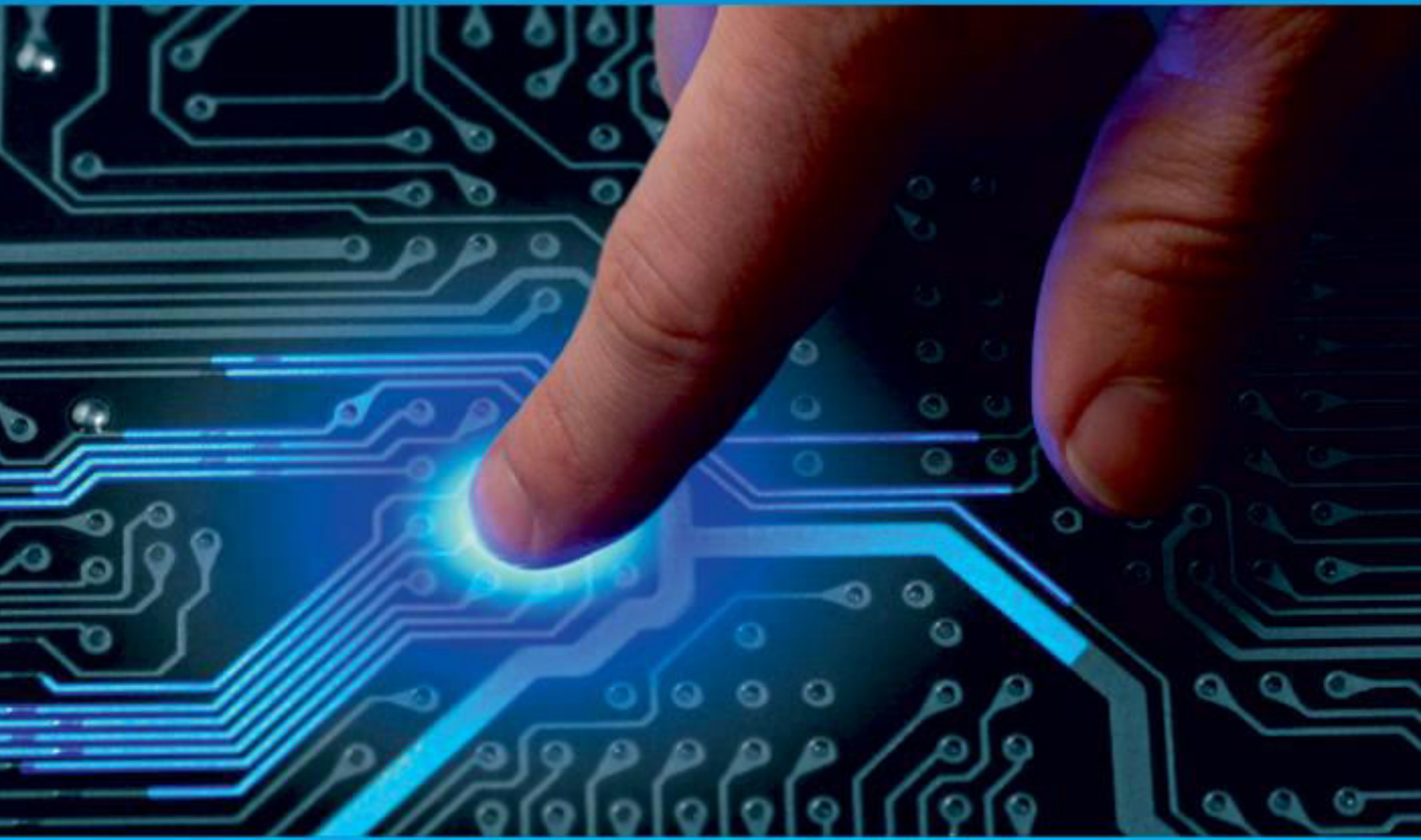
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 1, January 2022

ISSN
INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.542

9940 572 462 6381 907 438 ijircce@gmail.com www.ijircce.com

| DOI: 10.15680/IJIRCCE.2022.1001032 |

Comparative Analysis for Sarcasm Detection in Hindi Text using Three Approaches: Word Embedding, Context based Approach & Machine Learning

Maryam S. Siddiqui[#1], Sharvari S. Govilkar[#2]

Department of Computer Engineering, PIIT, New Panvel, Mumbai University, India[#1, 2]

ABSTRACT: Sarcasm is defined as special category of figurative language that means the opposite of what is said, made in order to hurt some one feelings or to criticize someone. Negative thoughts are expressed through positive intensified words. Today with the increase in communication on on social media such as Facebook, Twitter, WhatsApp, etc. negativity is expressed indirectly and this has become a new trend. Hindi is one of the popular Indian language that is highly used by Indians while communicating on social media . As Hindi language is rich in morphology and complex in structure, sarcasm detection is one of the tedious job. An extensive set of annotated training data to detect sarcasm is surely required. Availability of Hindi annotated dataset is almost negligible in the domain of sarcasm detection.Small amount of research has been carried in the field of sentiment analysis for Hindi Language. Information content in Hindi is important to be analyzed for the use of industries and government(s) as Hindi is considered to be the national language. The idea is to propose a system where the comparison between three approaches namely-Word Embedding ,Contextbased Approachand Machine learningalgorithm for Hinditext . Among three of them which approach works which higher accuracy and efficiency is seen .

KEYWORDS : Sentiment analysis, Sarcasm detection, Pos tagging, Pre processing,stop words, Tweets, Twitter,Word embedding, Context based approach Machine learning approach.

I. INTRODUCTION

Sentiment Analysis isidentifying and aggregating attitudes and opinions expressed by Internet users for a specific topic. In other words , it can be seen as attitude or opinion of individuals or society about a particular/current event/happening or topic. This process involves two steps a) information mining from various sources of text forms such as blogs, reviews and b) classification on basis of polarity as positive, negative or neutral. Sarcasm can be defined as to convey or express feelings where individual or society say or write something, that is completely different of what they actually intended or meant or said . When one uses sarcasmthe person speaks the contradictory of what the speaker means to express

gloomy feelings applying positive words. Retailers get to know the opinion, feedback of the customers through sarcasm. Sarcasm find its applications in many social networking sites and micro-blogging websites where users use positive intensified words to invade others which make use of sarcastic comments which in turn create problems for the individuals to say what it means.

Sarcasm is used for various purposes viz- criticism or mockery. Thus, identification ofsarcasm becomes hard even for humans to recognize. People make use of sarcasm generally to express their opinions or feelings especially in the social networking sites like Twitter and Facebook etc.in order toimprove accuracy of sentiment analysis perfect analyzing and understanding of the sarcastic sentences or statements is necessary.

To identify sarcasm in such natural Hindi tweets, the same feature set used for English scripted tweets might not be applicable efficiently. Therefore, one needs to rely on other parameters such as news context, specific patterns, rules, etc., to identify sarcastic Hindi tweets [4].

In the field of sentiment analysis for Hindi Language,small amount of research has been carried out till date. Hindi is considered to be the national language so information content in Hindi is important to be analyzed for the use of industries and government(s). Sentiment analysis is very challenging for Hindi language due to numerous reasons as follows:

(1) Lack of well annotated standard corpora, hence supervised machine learning algorithms can’t be applied.

(2) No efficient parser and tagger are present for this language as Hindi is a resource scarce language .
(3)Limited numbers of adjectives and adverbs are contained in limited resources like Hindi Senti WordNet (HSWN)available for this language. All the words in HSWN are present in inflected forms.Also, all the inflected forms of the word are not present in any of the resource.
(4) Even, Translation dictionaries do not / may not account for all the words reason being the language variations. Same words may be used in multiple contexts and context dependent word mapping is a difficult task, error prone and requires manual efforts.

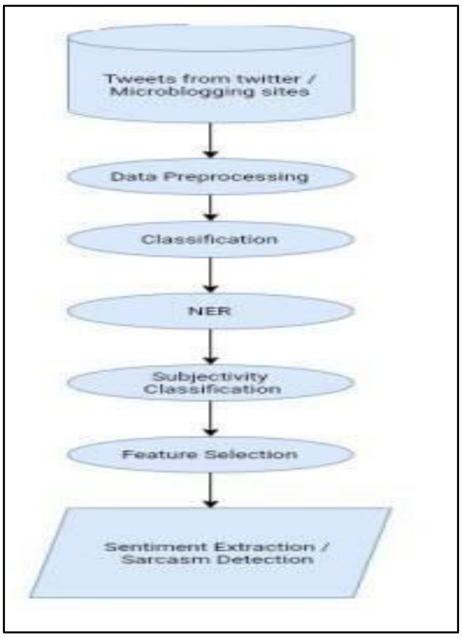

Tweets from twitter / Microblogging sites
Data Preprocessing
Classification
NER
Subjectivity Classification
Feature Selection
Sentiment Extraction / Sarcasm Detection

Fig 1: General process of Sarcasm detection

This paper proposes a system that accepts tweets from social networking site-Twitter as input ,data pre-processing takes place followed by use of various approaches ,further resulting into classifying the tweets as Sarcastic or non-sarcastic tweet. Related work and literature survey is discussed in section II. The proposed system is discussed in section III. Finally, section IV concludes the paper.

II. Literature survey

In this section we cite the relevant past literature that use the various approaches to detect sarcasm from various sources like Twitter, Facebook etc. Most of the researchers concentrate on English as a language rather detecting sarcasm in Hindi . Being morphological rich and due to lack of annotated data set, a little research has been carried out for Hindi as a language .

Abulaish et al. (2018) invokes the problem of self-deprecating sarcasm detection, a special case of sarcasm detection. Amalgamation of rule-based and machine learning approaches have been proposed for detecting self-deprecating sarcasm. Different categories of figurative language targeted were sarcasm, irony, satire, etc. in Twitter, but self-deprecating sarcasm detection was never considered.[1]
Aggarwal et al. (2020) studied a corpus of tweets for training custom word embeddings and a Hinglish dataset labelled for sarcasm detection was used. Deep learning-based approaches (including CNNs, LSTMs, Bi-directional LSTMs (with and without attention) are used to address the problem of sarcasm detection in Hindi-English code-mixed tweets using bilingual word embeddings derived from Fast Text and Word2Vec approaches. Attention based Bi-directional LSTMs gave the best performance exhibiting an accuracy of 78.49%.[2]

Code-mixed Hinglish tweets are fed as training, validation, and test sets for model training, validation,and testing respectively. NITS-Hinglish - Senti Mix
is an ensemble model wherein different models like basic LSTM (Long Short-Term Memory), LSTM + Convolution, a Bi LSTM (Bidirectional LSTM), and a CNN (Convolution Neural Network) model have been amalgamate to improve the general F-Score of the framework. [3]
Ilavarasan (2020)has done a survey about different sarcasm detection works done in past, given a brief about general architecture of sarcasm detection and various approaches used, different types of sarcasm, and some challenges in sarcasm detection. [10]

Sarsam et al. (2020) showed that using lexical, pragmatic, frequency, and POS taggingcontribute in improving the performance of SVM, and also lexical and personal features can enhance the performance of CNN-SVM.[16]

Bharti et al. (2018) proposes a pattern-based approach to identify sarcastic Hindi tweets where a set of online news is treated as temporal facts. [4]

Hazarika et al. (2018) adapted a hybrid approach of bothcontext-driven and contentmodeling for the purpose of sarcasm detection in online social media discussions such as Reddit called CASCADE (Contextual Sarcasm Detector). [9]

Swami et al. (2018)used English-Hindi code-mixed dataset of tweets for presence of sarcasm where each token was annotated with a language tag. A supervised classification system was developed using the same dataset that resulted in an average F-score of 78.4 after using random forest classifier and performing 10-fold cross validation.[18]

Bharti et al.(2018) approach followed in this research utilizes Hindi news as the context of a tweet within the same timestamp and obtained an accuracy of 87%.[5]

Sahaet al. (2107) provides classification based on the polarity of tweets as- positive, negative or neutral. In order to find accuracy of tweetsNaïve Bayes and SVM classifiers were used .[14]

Machine-learning algorithm and rule-based approach have been discussed in this paper. Sarcasm has different nature, shape and application in real life, hence it was found that this was a challenging problem and is too wide to be made as a generalized formula. [20]

Dave et al.(2016) various supervised classification techniques mainly used for sarcasm detection and their features have been identified. Results obtained from the classification techniques, on textual data available in various languages on review related sites, social media sites and micro-blogging sites have been analyzed.
Bouazizi et al. (2016) detected sarcasm on Twitter . A pattern-based approachhas been proposed with four sets of features that cover the different types of sarcasm. The tweets are classified as: Sarcastic and Non-sarcastic. An accuracy of 83.1% with a precision equal to 91.1% is by the proposed approached. [6]

Joshi et al. (2016)checked whether prior work could be improved using semantic similarity/discordance between word embedding or not.Four different types of word embedding have been experimented. Irrespective of the word embedding used or the original feature set to which our features are augmented appreciable improvement in sarcasm detection, had been observed. [12]

Zhang et al. (2016) investigates the use of neural network for tweet sarcasm detection, and comparison between the effects of the continuous automatic features with discrete manual features is made. [21]
Bouazizi et al. (2015) has proposed a method to detect sarcasm in Twitter that uses different components of the tweet. Four sets of features containing different types of sarcasm classified tweets into sarcastic and non-sarcastic.[7]

III. METHODOLOGY

The proposed methodology compares three different approaches–Word embedding, Context based approach and various machine learning approaches.

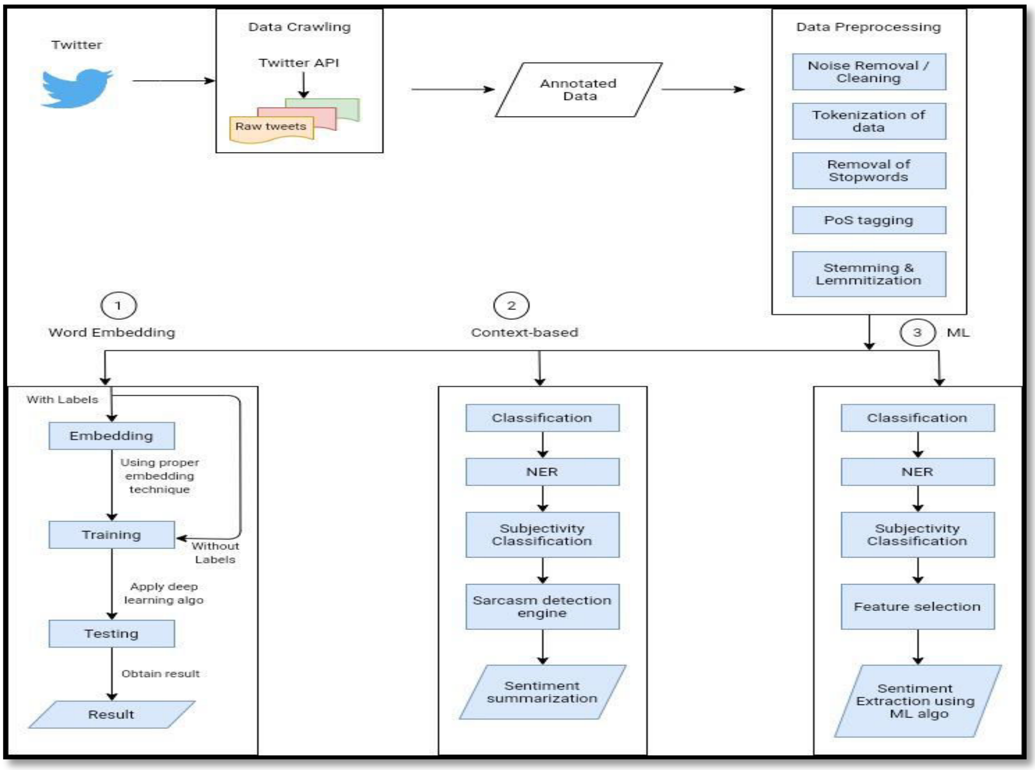


Fig 2: Proposed system architecture

As a result of Literature survey done it is found that very little work has been done on Hindi as a language due to lack of the appropriate corpus available. So, proposed approach will involve work on three different algorithms namely–Word embedding, Context based approach and machine learning algorithms. The initial step of collecting data from the source –Twitter, Annotation of data and data preprocessing will remain same for all the three approaches as discussed below. The output obtained from the data preprocessing will be given as input to all the three approaches and worked accordingly.

Step I: Data Crawling
The dataset is obtained from twitter. The obtained tweets are for Hindi language. Data crawler needs to be implemented using APIs.

Step II: Annotated data
This step includes annotation which search tags (hashtags)used for scraping the tweets. It is tokenized (manually) and classified. This will be our training set. The tweets are marked as having positive sarcasm or negative sarcasm. Tweets that have sarcasm or irony are marked to be withpositive sarcasm and tweets with hashtags like cricket, Bollywood have negative sarcasm.

Step III: Data preprocessing
The data obtained from any social media like Twitter, Facebook etc. has a lot of noise introduced into it which requires a lot of preprocessing. Data collected from twitter shows the presence of # tag and @ . Symbols # and @ needs to be removed. Removal of rare words (Words which have occurrence of words of less than 10 in the entire dataset) and search tags (like cricket and sarcasm) has to be done**.** URL's and punctuation marks are also removed. Data

preprocessing involves process such as removal of noise, tokenization of data, removal of stop words, Pos Tagging and stemming and lemmatization.

APPROACH I: WORD EMBEDDING:

Word Embedding in simple words are the texts which gets converted into number form and there may be various numerical representations of the same text. As many machine learning algorithms were incapable and almost all Deep Learning Architectures processed strings or *plain text* in their raw form, words embedding took birth. In order to perform any sort of job, Word embeddingrequire numbers as inputs be it classification, regression etc.A Word Embedding does the work of mapping word using a dictionary to a vector.

Step I: Data Crawling
Step II: Annotated data
Step III: Data pre- processing
Step IV: Embedding

The two types of embedding which we can try upon for Hindi as language are:

a) Fast Text: Fast Text is an open-source, free, lightweight librarycreated by the Facebook Research Team for efficient learning of representations of words and sentence classification.Fast Text is different from word2vec every single word treated as the smallest unit whose vector representation is to be found. In Fast Text a word is formed by a n-grams of character.Consider an example - Sunny is composed of [sun, sunn,sunny],[sunny,unny,nny] etc., where n ranges from 1 to the length of the word.

Forexample, for a word like *stupedofantabulouslyfantastic,* which might never have been in any corpus, the result might return any two of the following solutions – a) a zero vector or b) a random vector with low magnitude. Fast Text however can produce vectors better than random by breaking the above word in small chunks. Further making use of the vectors for those chunks to create a final vector for the word. As a result the final vector might be closer to the vectors of word fantastic and fantabulous.

a. character n-grams embedding tend to perform better than word2vec and glove on smaller datasets.

b)Bidirectional Encoder Representations from Transformer (BERT)

BERT involves a method of pre training language representations . BERT produces word representations that are dynamically informed by the words present around them. For example, consider these two sentences:"The man was accused of robbing a bank." "The man went fishing by the bank of the river." Word2Vec shall produce the same word embedding for the word "bank" in both cases, while under BERT the word embedding for "bank" would be different for each sentence.

Step V: Using deep learning models for testing and training followed by obtaining of results

Deep learning models uses several algorithms. No one network is considered perfect, some or the other algorithm is best suited for a specific task as per the requirement.

a)Convulation Neural Network

CNN's, consists of multiple layers and are mainly used for image processing and object detection. CNN's area also used in identifying satellite images, process medical images, forecast time series, and detect anomalies. CNN's consists of multiple layers that process and extract features from data.

b) Long Short-Term Memory Networks (LSTMs) Recalling past information for long periods is the default behavior of LSTMs.They work in the following way:

- First, they forget irrelevant parts of the previous state
- Next, they selectively update the cell-state values
- Finally, the output of certain parts of the cell state

c) Recurrent Neural Networks (RNNs)

RNNs have connections that form directed cycles which allows the outputs obtained from the LSTM to be fed as inputs to the current phase in RNN. The output which is obtained from the LSTM becomes an input to the current phase and has the capability to memorize previous inputs due to its internal memory.

APPROACH II : CONTEXT BASED

a) News and Tweet Collection

In order to extract important keywords of all the collected and processed news from news corpus, an algorithm has to be used .This algorithm will find POS tags information and check the presence of any proper noun, verb and noun if there. If such tags are found, then the corresponding keywords append to the set of keywords. The at most purpose of this algorithm is to extract the subject, object and verb of the news as important keywords. The end result of this algorithm is the tags which will be obtained and are assumed to be - proper noun (NNP), verb (V), and noun (NN) will work as subject, verb, and object, respectively.

b) Tweet Annotation

The data annotation is done by some practitioner or teachers who are professionals.

c) Sarcasm Detection Engine

In Sarcasm detection Engine two work will be carried out- Sentiment identification and Context identification.

i) Sentiment identification

Here identification of sentiment of a tweet is done where sentiment value can be either positive, negative or neutral.

ii) Context Identification in Tweet and News:

This algorithm takes Tweet Corpus (C1), News Corpus (C2) and Hindi Senti WordNet (HSWN) as input and gives the context polarity of tweet and news using corpus as output. It does that by POS tagging information and extracts context phrase using rule set given in Algorithm. Further, the polarity value of each phrase is found .

d) Sarcasm Detection Algorithm

If the context polarity of a tweet is contradicted with context polarity of the related news the tweet will be classified as sarcastic. In other case, if the context polarity of a tweet and related news is same, but the sentiment of the tweet is contradicted then the tweet is classified as sarcastic. Otherwise, the tweet will not be classified not sarcastic.

APPROACH III: MACHINE LEARNING

The initial three steps will remain same for this approach as well. The output of the data preprocessing will be input given to different machine learning algorithms.

Step I: Data Crawling

Step II: Annotated data

Step III: Data pre- processing

Step IV: Classification

This is an essential step as data will have different set of features for different domains and thus, each domain should have different classifier.

Step V: Named Entity Recognition

NER is considered to be the most important part of sentiment analysis process. When an opinion document is considered, all quintuples namely - entity, aspect, sentiment on aspect of the entity, opinion holder and the time/context of opinion should be taken into account .

Step VI:Subjectivity Classification –Here, the sentence is classified as subjective or objective - subjective sentences have sentiments and objective sentences contains facts and figures.

Step VII: Feature Selection - The features selected can be unigrams and/or bigrams or higher n-grams with/without punctuation and with/without stop words with presence (Boolean)/count(int)/tfidf (float) as feature scorer for each sentence/paragraph/file.

Filtering of stop wordsreduces accuracy. Adverbs and determiners that start with "wh" can prove to be valuable features.A dip in accuracy can be caused by removing stop words . On the same basis, punctuation helps in detecting sarcasm and exclamation, hence should not be eliminated from the tweets or sentences taken as input.

Step VIII: Sentiment Extraction using Machine learning Algorithm

a) SVM

"Support Vector Machine" (SVM) is a supervised machine learning algorithm that can be usedfor both classification or regression .According to this algorithmdata item are to be plotted as a point in n-dimensional space where the value of each feature is the value of a particular coordinate. Next
classification is performed by finding the hyper-plane that differentiates between the two classes very well.

b) NAÏVE BAYES ALGORITHM

It is a classification technique based on Bayes' Theorem . In simple terms, a Naive Bayes classifier makes an assumption that the a particular feature present in a class is unrelated to the presence of any other feature.Naive Bayes model is easy to build and majorly useful for very large data sets. Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c).

c) RANDOM FOREST

Random Forest is a supervised machine learning technique used for both Classification and Regression problems. It is based on the concept of **ensemble learning,** that involves process of combining multiple classifiers in order to solve a complex problem and to improve the performance of the model.

d)k-Nearest neighbors (KNN)

K-Nearest Neighbor is a supervised Machine Learning technique algorithm .This algorithm considers the assumption of the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and provides a classification of a new data point based on the similarity.

IV. CONCLUSION

Sentiment analysis is the way to find one's opinion towards any specific target. Sentiment Analysis is a method in which a piece of writing is positive, negative or neutral is determined.Sentiment analysis is helpful in various fields such as data analysts within large enterprises gauge public opinion, conduct nuanced market research, monitor brand and product reputation, and understand customer experiences. Sarcasm detection is one of the major applications of Sentiment analysis. In today's world, posting of sarcastic messages on social media is very common. Sentiment analysis has become a challenging task due to the presence of sarcasm in its Sarcasm expresses feelings in which people say or write something which is actually different from what they intend to say. Sarcasm detection in Hindi is a tedious job due to its richness in morphology and complexity in structure. Lack of efficient annotated corpora for Hindi language has left it less explored.
Existing researchers have introduced various approaches to find sarcasm in a given document/tweet. For comparing various approaches namely – Word embeddings, Context based, Machine learning approaches have been considered. The various approaches named have been applied on English language so this creates a further scope of applying the same approaches on Hindi language and comparing which among all the three works efficiently and accurately.

REFERENCES

[1] Abulaish, M., & Kamal, A. (2018, December). Self-deprecating sarcasm detection: an amalgamation of rule-based and machine learning approach. In 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI) (pp. 574-579). IEEE.

[2] Aggarwal, A., Wadhawan, A., Chaudhary, A., & Maurya, K. (2020). " Did you really mean what you said?": Sarcasm Detection in Hindi-English Code-Mixed Data using Bilingual Word Embeddings. arXiv preprint arXiv:2010.00310.

[3] Baroi, S. J., Singh, N., Das, R., & Singh, T. D. (2020). NITS-Hinglish-SentiMix at SemEval-2020 Task 9: Sentiment Analysis For Code-Mixed Social Media Text. arXiv preprint arXiv:2007.12081.

[4] Bharti, S. K., & Babu, K. S. (2018). Sarcasm as a contradiction between a tweet and its temporal facts: a pattern-based approach. International Journal on Natural Language Computing (IJNLC) Vol, 7.

[5] Bharti, S. K., Babu, K. S., & Raman, R. (2017, December). Context-based sarcasm detection in hindi tweets. In 2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR) (pp. 1-6). IEEE.

[6] Bouazizi, M., & Ohtsuki, T. O. (2016). A pattern-based approach for sarcasm detection on twitter. IEEE Access, 4, 5477-5488.

[7] Bouazizi, M., & Ohtsuki, T. (2015, December). Sarcasm Detection in Twitter:" All Your Products Are Incredibly Amazing!!!"-Are They Really? In 2015 IEEE Global Communications Conference (GLOBECOM) (pp. 1-6). IEEE.

[8] Dave, A. D., & Desai, N. P. (2016, March). A comprehensive study of classification techniques for sarcasm detection on textual data. In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (pp. 1985-1991). IEEE.

[9] Hazarika, D., Poria, S., Gorantla, S., Cambria, E., Zimmermann, R., & Mihalcea, R. (2018). Cascade: Contextual sarcasm detection in online discussion forums. arXiv preprint arXiv:1805.06413.

[10] Ilavarasan, E. (2020, March). A Survey on Sarcasm detection and challenges. In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS) (pp. 1234-1240). IEEE.

[11] Jain, T., Agrawal, N., Goyal, G., & Aggrawal, N. (2017, August). Sarcasm detection of tweets: A comparative study. In 2017 Tenth International Conference on Contemporary Computing (IC3) (pp. 1-6). IEEE.

[12] Joshi, A., Tripathi, V., Patel, K., Bhattacharyya, P., & Carman, M. (2016). Are word embedding-based features useful for sarcasm detection? arXiv preprint arXiv:1610.00883.

[13] Mittal, N., Agarwal, B., Chouhan, G., Bania, N., & Pareek, P. (2013, October). Sentiment analysis of hindi reviews based on negation and discourse relation. In Proceedings of the 11th Workshop on Asian Language Resources (pp. 45-50).

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

[15] Sana, P., & Avinash, S. (2017). Opinion Mining in Twitter: How to make use of Sarcasm to Enhance Sentiment Analysis. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 6.

[16] Sarsam, S. M., Al-Samarraie, H., Alzahrani, A. I., & Wright, B. (2020). Sarcasm detection using machine learning algorithms in Twitter: A systematic review. International Journal of Market Research, 1470785320921779.

[17] Sharma, A. Hindi Text Emotion Recognition based on Deep Learning.

[18] Swami, S., Khandelwal, A., Singh, V., Akhtar, S. S., & Shrivastava, M. (2018). A corpus of English-Hindi code-mixed tweets for sarcasm detection. arXiv preprint arXiv:1805.11869.

[19] Wang, Z., Wu, Z., Wang, R., & Ren, Y. (2015, November). Twitter sarcasm detection exploiting a context-based model. In international conference on web information systems engineering (pp. 77-91). Springer, Cham.

[20] Wicana, S. G., İbisoglu, T. Y., & Yavanoglu, U. (2017, January). A review on sarcasm detection from machine-learning perspective. In 2017 IEEE 11th International Conference on Semantic Computing (ICSC) (pp. 469-476). IEEE.

[21] Zhang, M., Zhang, Y., & Fu, G. (2016, December). Tweet sarcasm detection using deep neural network. In Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers (pp. 2449-2460).

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details