RESEARCH

Detection of Political Intent through Analysis of Tweets and Homophily Elements

Nisha Shetty[1], Balachandra Muniyal[1*], Daita Ravi Teja[1] and Leander Melroy Maben[2]

Abstract

Twitter is one of the most popular social networking platforms of today's generation and is a fundamental tool in harvesting the data of many users worldwide. Many discussions ranging from current affairs, news sharing, filing complaints to advertising and discussing some common interests etc. happen on Twitter. It is widely used by many famous politicians as a prominent communication medium to address large masses ,owing to its mass usage and popularity . Thus, it is safe to assume that people communicate their political ideologies on Twitter. Many works have been done so far to deduce the stance of user towards a particular party by performing sentiment analysis on their tweets using popular classifiers. To find connected and similar users, earlier works generated a social network graph, based on the assumption that friends and followers share similar interests, which might not be true in all cases. In contrast, the proposed work employs the concept of ensemble classifier (a single classifier generated from several base learning classifiers) to analyze the tweets and makes use of multiple interaction elements like followers/ following, mentions, re-tweets ,hash tags etc. to infer which political party a user identifies with. These interaction elements project out the homophily (users who share same beliefs and choices) amongst the users. The proposed study can be encompassed to any domain and can be used by advertising agencies, marketing companies, e-commerce, heath care etc. to identify their target audience.

Keywords: homophily; Online Social Networks; CNN; LSTM; Ensemble; Sentiment Analysis; Sarcasm Analysis

1 Introduction

Social networks have become a crucial part of today's generation for information and opinion exchange. One such popular site is Twitter with more than 1.45 billion registered users and 330 million monthly active users tweeting an approximate of 500 million tweets per day [1]. Twitter is the most common micro-blogging platform in countries like the United States, Japan and India. Its popularity has encouraged the researchers to analyze its contents and detect hidden patterns.

In recent years, Twitter has become a popular campaigning tool and medium of communication between the leaders and voters. Many prominent political leaders are very active on Twitter, and with lots of political discussions/opinions on-board, makes Twitter an authentic platform to predict a person's emotion and his political affinity.

*Correspondence: bala.chandra@manipal.edu
[1]Department of Information and Communication Technology, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India
Full list of author information is available at the end of the article

Data present in Online Social Network (OSN) can be modelled as G= (V,E) where V embodies actors (people , groups etc.) and E symbolizes ties or relationships between the nodes [2]. Fig. 1 gives a glimpse of social network in the graphical form.

Data in Online Social Networks [3] [4] conveys the following:

- Profiles: Give the portrayal of user/alter-ego to the outside world.
- Connections: Showcase the relationship between users like friendships, follow, following etc.
- Messages/Tweets/Hashtags: Include the interac- tions amongst users/groups.
- Multimedia: Collective terminology for audio, video and pictures exchanged or uploaded by the users.
- Tags/Mentions: Include the metadata linked to a particular content.
- Preferences: Consist of likes and causes supported which can be explicitly stated or hidden

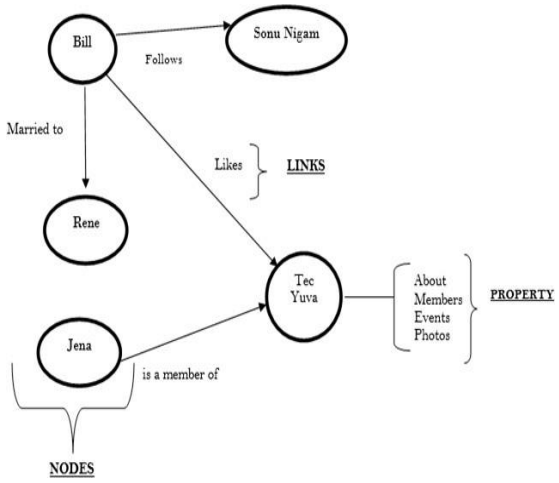

Figure 1 Graphical Representation of OSN [2]

- Behavioral information: Can be inferred from user activities in such sites.
- Login credentials: Ensure that the use is restricted to authorized users.

Sentiment Analysis [5] ,a popular Natural Language research area , is extensively used in many domains such as brand exploration, vote share prediction, current affair news analysis etc. User generated content in such online platforms are utilized by many advertising companies to market their products thereby improving their sales (through feedbacks).Generally machine learning classifiers [6] like Na¨ıve Bayes have shown better accuracy when compared to the lexicon based approach [7].

Two most common psychological traits observed in social media are Homophily and Social media influence. Homophily is the tendency of the users to interact with analogous minded people having similar likes. Social media influence highlights upon the fact that attitudes of the people get affected by their peers in their social circle [8].

Along with the tweets our research analyses the communication elements of user such as follow/following, mentions, retweets, hashtags etc. to gather similar users together.

The proposed methodology is two-fold process wherein the first part analyzes the tweets of the user using an ensemble [9] and deep learning classifier [10] to predict the political party of a user. This is supplemented with the analysis of tweets and sarcastic contents, if present. In the second part the similar users are grouped together using various interaction elements (retweets, mentions, following/followers, hashtags etc.) [11] [12] and machine learning classifiers are employed to determine the notion of user towards a particular group (positive/negative).

The rest of the paper is organized in the ensuing fashion. Section 2 touches upon the the prominent and latest research in the constituent parts of this study. Detailed introspection of the proposed methodology is presented in section 3 . Section 4 exhibits the obtained results with suitable error analysis. The paper is concluded with some discussions on possible future work in section 5.

2 Related Works

The proposed work can be partitioned into 3 major domains on which many prominent works are done. Some of the referred recent works are listed below.

2.1 Sentiment analysis & Prediction of chosen political party.

Any Sentiment Analysis approach falls into the following 2 broad categories:

- Lexicon / Corpus based approach: Here the sentiments are analyzed by referring to any popular dictionary to detect schism of the opinion word.
- Machine Learning based approach: Prominent classifiers are pre-trained and subsequently tested to determine the predilection of the tweet.

Sharma et al. [13] (2018) enunciated the stance of people towards 2 popular parties Conservative and Labour. k-NN offered the best accuracy for their dataset and authors observed a rise in popularity of Labour party and a decline in that of Conservative party.

Widodo Budiharto and Meiliana Meiliana [14] (2018) applied sentiment analysis to predict the popular political leader among the Indonesians. The polarity of the tweets were decided based on total number of positive and negative words for a particular leader. This was augmented with likes and retweets favoring a leader. We have improved upon this work by adding more user signals in our research.

Alaoui et. al. [15] (2018) considered the semantic meaning of the tweets and developed their own labelled corpus using prominent hashtags used per party. Elongated words were given different priority scores and influence of a tweet was highlighted by the incorporation of retweets and likes (More retweets/likes equals more preference towards that party).The comparison of the proposed algorithm against Google Cloud prediction API and Na¨ıve Bayes showed favorable results. However, as quoted by the authors themselves, increasing their sample size can substantially improve their work. An interesting application involving deep learning approach for personality detection was observed in [16]

which emphasized on processing the semantic information in user generated texts. Xue et al. (2018) governed the linguistic digits of the tweets (no. of punctuations, capitals, text frequency, etc.) along with inferring the psychological aspects of a complete tweet set per user to detect the major emotion projected. It was observed that the proposed CNN based detection system improved the prediction accuracy of other regression models.

Nagarajan et al. [17] (2018) improved the classification accuracy of the decision tree by increasing the apt feature set using particle swarm optimization technique and genetic algorithm. They were successful in classifying emotions into multiple classes and their hybrid technique offered better detection than other base classifiers. The authors hope to upgrade upon their work so as to improve on the optimization.

Abhishek Kumar et al. [18](2019) incorporated a bidirectional LSTM which studies the context of a word in a sentence and ultimately identifies the sentence polarity. This two level approach first studied the word meanings through Distributional Thesaurus and ultimately applied it to the sentence to detect opinions and emotions. However, the error analysis of their approach showed that in compound sentences, many differing emotions tend to overwhelm the classifier often leading to misclassification.

Wang et al. [19] (2019) surveyed on various techniques for stance detection. They categorized their study into 2 main domains; one with textual content only and other where textual content was supplemented with metadata. Also they studied various works on feature extraction and opinion mining at various levels (corpus/document). They further highlighted on the challenges faced in online opinion mining and offered some possible solutions. They indicated the use of machine learning models which work on base data to increase the sample size. A move towards target specific attention based network for research was suggested.

Hima Suresh et al. [20] (2019) put forth an inventive approach for sentiment analysis by modifying the C4.5 decision tree which outperformed all the other classifiers by a significant margin.

Ankita Sharma and Udayan Ghose [21] (2020) have analyzed the Twitter data in context of the general elections in India. In their study, the packages of R have been used to collect and pre-process the data. R and Rapid Miner's Alyien were incorporated to check the sentiment tallies (for or against), polarity and subjectivity of the tweet. Since the study was limited to the exploration of tweets, it can be extended to use multimedia, hashtags etc. Also a striking limitation observed was the while collection of tweets the location was not generalized causing bias in the study.

Mohd Zeeshan Ansari et al. [22] (2020) analyzed the political scenario in India via tweet analysis. Although the majority of the classifiers (LSTM, Decision Tree, Random Forest, SVM and Logistic Regression) used in their work matches with ours, labelling with human annotators did not get them a reasonable corpus. Future work in this domain would be using semi-supervised classifiers to incorporate better reservoir (which includes more jargons and slangs) so as to improve the training and analysis of the classifiers.

2.2 Stance & Homophily Detection

Rossetti et al. [23] (2016) investigated the relationship between topological features of various networks and the degree of homophily between their subscribers. Skype, Last FM and Google+ users were inspected to determine similarity in usage pattern, listening activity and education respectively. Users were grouped using various community detection algorithms and new users were classified to appropriate groups based on their chronological, terrestrial and topographical features. Like us, they further plan to explore the strength of bonds between users to endorse better prediction.

Darwish et al. [24] (2017) put forth the analysis of interaction elements to detect homophily. The proposed similarity formula was able to amass the people sharing the same stance together. Their study yielded a good result for retweets / hash but not for a bunch of interaction elements together. Nonetheless, the authors speculated the reason for it being improper analysis of URLS.

Du et al. [25] (2017) proposed a novel neural network based stance detection algorithm called Target specific attention neural network which incorporates the following:

- A RNN to extract object specific data.
- LSTM and attention mechanism to muse on prominent parts of the texts with respect to the object and use the information articulated to detect stance.

Further authors aim to find a suitable way to add external knowledge to improve the accuracy. They also plan to combine their approach to other machine learning algorithms.

Rajendran et al. (2018) [26] compared the accuracies of Grated Recurrent Unit (GRU) and Bidirectional Long Short-Term Dependency (LSTM) to detect the stance of the user on news data set crawled from popular news media. Bidirectional LSTM proved to offer better detection and we incorporated the parameters they used in our work.

Poddar et al. [27] (2018) employed CNN and RNN based encoders to determine the hidden stances in users' tweets. Many user' stances on a particular topic

were then aggregated to check if it's a rumor or not. This model when combined with authenticity detection using transfer learning offered better results than its other contemporaries.

Barone and Coscia [28] (2018) investigated the tax fraud in business partnerships by considering users as nodes and versions about them as edges. They gave a new take on trust computation and analyzed how trust score and similarities amongst groups tend to contribute to the homophily.

Mirko Lai et al. [29] (2019) investigated the opinion of the user on the reforms introduced in Italian Constitution by effectively analyzing the tweets and network elements such as retweets, mentions, reply-to, follow/following relationships. We in our research also reached the same conclusion as proven by them that retweets, mentions, follow/following can exhibit homophily. They established the fact that difference of opinion could be expressed using reply-to which can be incorporated in our future analysis. In future, the authors aim at studying the influence of a prominent person/bot in changing the discernment of his peers.

Sailunaz and Alhajj [30] (2019) generated their own data set where tweets and replies were analyzed for the sentiments and emotions expressed. The user's influence was further determined using number of followers, retweets, likes etc. The authors could generate a customized recommendation system with their approach. However, their experiments focused on simple texts tailor-made for their approach and it needs to be improved to deal with random abbreviations, emoji, hashtags etc. prominent in any open platform.

Abeer Aldayel and Walid Magdy [31] (2019) categorized Twitter data into likes, interaction (mention, retweet, and replies), connection (follow) and textual contents. The stance was analyzed using linear SVM which offered good results. One of the major shortcomings of this method was it indicated that the presence of retweet, reply, mention as support which need not hold true always. The proposed classifier often misclassified the 'none' stance into either of the classes owing to biased and constrained dataset.

Hamdi et al. [32](2019) employed node2vec to extract important features from user follower/following and combined with user features extracted via Twitter APIs to detect fake news in Twitter. The extracted features were used as a metric to vouch for user credibility. In future, the authors plan onto extend the proposed methodology to detect rumors, junks etc. while cashing on the opportunities in domains such as recommendation systems.

Darwish et al. [33] (2020) exhibited unsupervised learning algorithm to categorize the stance of the users. Preprocessed tweets underwent dimensionality reduction to reduce the effect of outliers prior to clustering via DBSCAN and mean shift. The labeling time of the stances was significantly reduced when compared to supervised learning approaches. However, their method was limited only to tweets and can be extended to incorporate all other metadata/multimedia.

2.3 Emoji & Sarcasm Analysis

Fede et al. [34] (2017) scrutinized the emoji statistics usage on various social networking platforms and grouped them into 3 categories page rank, popularity and simultaneously used emoticons. The authors were able to deduce sarcasm by consequent conflicting emoticons (positive emoji followed by a negative one or vice versa.). They could model most common emoji used per subject (area of interest). They further plan onto explore semantic information conveyed if emojis are used as a language.

Chen et al. [35] (2018) explored the usage of a particular emoji in both positive and negative context. Further, their proposed attention based LSTM amalgamates this bi-sense embedding scheme to predict the sentimentalities in a better fashion, when compared to the prevailing techniques. In future, the authors intent to espouse multi-sense embedding too.

Subramanian et al. [36] (2019) worked on the research gap of ambiguities experienced in sarcasm detection owing to small texts in social media. In their approach sarcasm detection was done via texts and emoji used. Our work is inspired by the methodology followed by then to infer sarcasm from emoticons.

Li et al. [37] (2019) improved the sentiment analysis of Weibo tweets by integrating in depth emoji analysis with attention based GRU network which focuses on key words in a sentence. The emoji were classified into positive, negative and humorous. Misclassification occurred due to ambiguous usage of some emoticons and limited data set. The authors plan onto incorporate machine learning approaches / image processing to improve the prediction capability of their classifier.

Cai et al. [38] (2019) expanded sarcasm detection to include image and textual attributes in detection. LSTM was used to infer apt textual features and the model developed by the authors extracted image traits to reckon if the tweet is sarcastic or not. The model failed to give appropriate classification when the attributes fetched from the text and images were contradictory. The authors hope to improve their work by including background knowledge and extend it to include audio visuals.

Potamias et al. [39] (2020) engineered a transformer based network architecture in order to eliminate the huge computation costs involved in data preprocessing as observed with the conventional methods.

Their RoBERta architecture which was an hybrid deep learning architecture involving both CNN and RNN (used to grasp circumstantial and temporal information) proved effective by a large margin in detection of figurative language, when compared to other state of art techniques.

Khotijah et al. [40] (2020) applied Paragraph2vec to find the context in the tweets and then classified if the tweets were sarcastic or not using LSTM. Their method worked well for Indonesian tweets, but was found deficient owing to sparse dataset in case of English tweet. Improvement can be done to scrutinize the expressed emotions in various contexts to improve the detection rate.

Sundararajan et al. [41] (2020) studied how user's mood change affects the level (polite, rude, rampant, and deadpan) of sarcasm expressed. Feature set for classification was gathered using ensemble classifiers and the strength of sarcasm was assessed with respect to positive/negative word count .Intensifiers used were evaluated using a fuzzy logic based approach. Tweets of a user over a time period was scrutinized to objectify his mood variations. The only limitation, we garnered is that the tweets were ranging over a wide span of topics, and were not contextual as exhibited in our study.

3 Metholology

The proposed methodology incorporates the following steps as shown in Fig. 2:

- The data was collected from prominent Twitter handles and existing data set from Kaggle [42] was also used.
- Tweets were analyzed to find the opinions of the users towards 'democratic' or 'republican' groups.
- Sarcasm detection using Convolution neural network was done to enhance the sentiment analysis procedure.
- Interaction elements of the users (friendships, retweets and the textual traits (hashtags and mentions) were accessed to predict their political affinity.

Following subsections explain the above mentioned steps in detail.

3.1 Data Collection

An ample list of prominent party handles , active party leaders's Twitter handle names, election campaign handles , common hashtags and search keywords of parties were prepared. Geo Tagging was applied to set the location to the United States. Kaggle data set was used as a secondary source to supplement our data set. Some of the popular keywords and handles used were *'Democrats', 'Republican', 'Donald Trump', 'Hillary Cinton','@realDonaldTrump','@HillaryClinton'*. Further using some seed users secondary user profiles were fetched from their network as shown in the Fig. 3 [8].

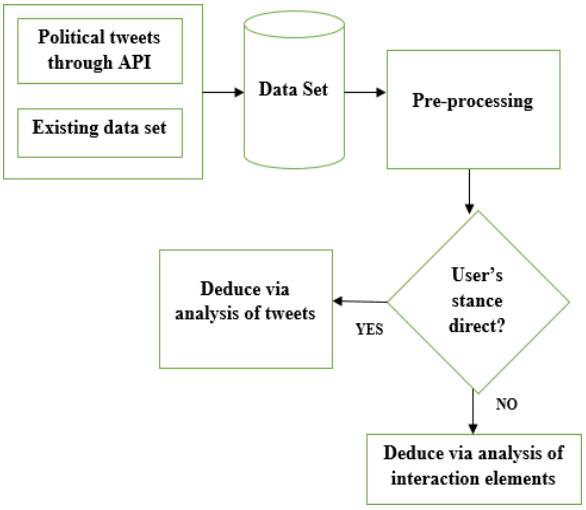

Figure 2 Overall Methodology

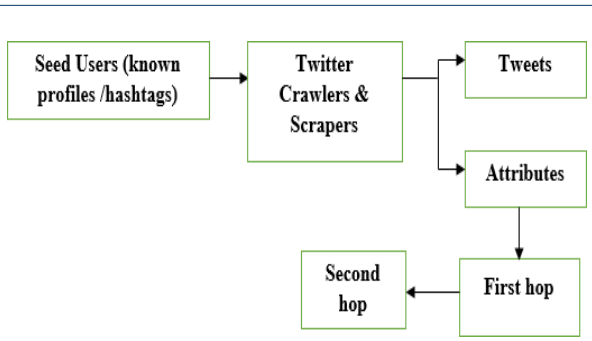

Figure 3 Data Collection

3.2 Data Cleaning

Algorithm 1 details the steps followed [43] [44] [45] [46] [47] [48] [49] :

Algorithm 1 Data Cleaning

for *each tweet i* **do**
  remove duplicate tweet, if any
  remove special characters, extra white spaces and punctuations
  oust stop words, non English text and URLs
  convert the tweet to lowercase
  substitute emojis, slang words and other common Twitter dialects with their actual meanings
  expand/shorten short forms (ex. DT => Donald Trump) /elongated words (ex. looooooooove =>love) to their actual format
end
return $cleaned - tweet_i$

3.3 Sentiment Analysis

The process is illustrated in Fig. 4 and explained in Algorithm 2.

3.3.1 Pre-processing [50] [51]

- Words were reduced to their root words.
- Word net corpus was used to lemmatize the words (ex. best is equivalent to good) to its root word.
- Using Tokenizer Library sentence was split into its constituent parts to obtain the opinion words (usually the adjectives/adverbs). Negation words such as not were also identified.
- Vectorization using Term frequency-inverse document frequency (TFIDF) was done to obtain the rare and frequent words used with respect to a party [52] [53].
- Bag of words of each of the parties were protracted.
- Labels of each party were converted to integer format to aid in classification process.

3.3.2 Applying sentiment analysis to the tweets.

Following classifiers are used for the process.

1. Multinomial Na¨ıveBayes (Probabilistic Classifier) [54]
2. Random Forest (Rule Based Classifier) [55]
3. Support Vector Machine (SVM) (Linear Classifier) [56]
4. Bidirectional Long Short Term Memory (LSTM) (Deep Learning Classifier) [57] [58] [59]
5. Ensemble of Na¨ıveBayes , Random Forest and SVM

Algorithm 2 Sentiment Analysis Explained

for *each sentence s* **do**
  split the sentence into its constituents
  find the target word
  extract the adjective/ adverb as the opinion word and add it to adjectivelist.
  for *each adverb/adjective* w_i *in the adjective-list.* **do**
    if *the word* w_i *is present in the seed-list s* **then**
      d seed list is any popular dictionary used
      identify its proclivity (positive/negative)
    end
    if w_i *has a synonym* s_i *in the seed list s* **then**
      w_i's inclination = s_i's inclination
    end
    if w_i *has a antonym* s_i *in the seed list s* **then**
      w_i's inclination = opposite of s_i's inclination
    end
    if w_i *is preceded by any negation word such as not* **then**
      w_i's inclination = opposite of w_i's inclination
    end
  end
end

Algorithm 3 and Algorithm 4 [60] brief the processes carried out in construction of Ensemble and subsequent political party inference procedure.

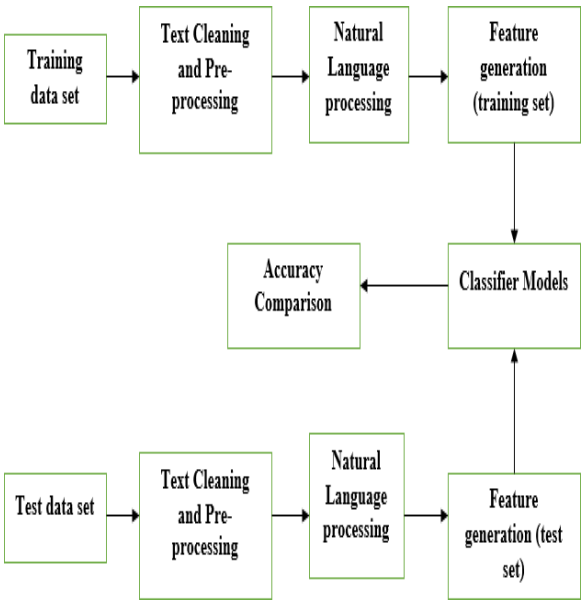

Figure 4 Sentiment Analysis

Algorithm 3 Algorithm for Ensemble

Input : Tweets **Output:** political party score

foreach *Tweet i* **do**
  $pos - democrat_i \leftarrow 0$ $pos - republic_i \leftarrow 0$
  $neg - democrat_i \leftarrow 0$ $neg - republic_i \leftarrow 0$
  /* initialization */
  foreach *base classifier* c_i **do**
    if c_i *predicts positive sentiment for a party p* **then**
      $pos\_p_i \leftarrow pos\_p_i + 1$
    else
      $neg\_p_i \leftarrow neg\_p_i + 1$
  /* where p can be democrat / republic */
  end
  $probability(pos - democrat_i) = \frac{pos-democrat_i}{pos-democrat_i+neg-democrat_i}$
  $probability(pos\_republic_i) = \frac{pos-republic_i}{pos-republic_i+neg-republic_i}$
  $probability(neg - democrat_i) = \frac{neg-democrat_i}{pos-democrat_i+neg-democrat_i}$
  $probability(neg\_republic_i) = \frac{neg-republic_i}{pos-republic_i+neg-republic_i}$
  $totalprobability(democrat_i) = probability(pos\_democrat_i) + probability(neg\_republic_i)$
  $totalprobability(republic_i) = probability(pos\_republic_i) + probability(neg\_democrat_i)$
end
foreach *base classifier* c_i **do**
  $weightc_i = \frac{accuracy\ of\ c_i}{Summation\ of\ accuracy\ of\ all\ classifiers}$
end
foreach *Tweet i* **do**
  $democrat - score_i \leftarrow 0$ $republic - score_i \leftarrow 0$
  /* initialization */
  foreach *base classifier* c_i **do**
    if c_i *predicts Democratic* **then**
      $democat - score_i = weightc_i * totalprobability(democrat_i)$
    else
      $republic - score_i = weightc_i * totalprobability(republic_i)$
    end
  return $republic - score_i$, $democrat - score_i$
end

Algorithm 4 Algorithm to find the party the user prefers.

Input : $Tweet_i$, rs_i, ds_i,$user_j$ **Output:** political party
if $rs_i > ds_i$ **then**
  politicalparty = "REPUBLICAN"
else
  politicalparty = "DEMOCRATIC"
/* where rs_i and ds_i are Democrat and Republic score returned in Algorithm 2 */
foreach *user j get all the political related tweets using cosine distance formula* **do**
  if $trs_j > tds_j$ **then**
    politicalparty = "REPUBLICAN"
  else
    politicalparty = "DEMOCRATIC"
  /* where trs_j and tds_j are sum of total Democrat and Republic scores of a $user_j$ computed via Algorithm 2 */
  $cos(t1,t2) = (t1 \bullet t2) / (||t1|| \bullet ||t2||)$;output =1 means high similarity.
end

3.4 Sarcasm Analysis

To enhance the accuracy of sentiment analysis, sarcasm exploration is also incorporated. Sarcasm is an ironic way to convey contempt and is often misleading (subjective to a person).Normally, sarcasm is used to appear funny, show angst and evade giving a straight answer. Our model labels sarcastic tweets towards "Democratic" as a preference towards "Republican" and vice versa. Algorithm 5 provides an insight on the procedure followed.

Algorithm 5 Sarcasm Analysis Explained

for *each sentence i* **do**
  split the sentence into its constituents
  collect the verbs and the adjective/adverbs.
  check if there is an inconsistency in the orientation of verb and adjective (positive verb with negative adjective and vice versa.)-> condn1
  check if the subsequent sentence conveys a opposing sentiment.-> condn2
  emotions and hashtags are analyzed for sarcasm contents.-> condn3
  presence of common sarcastic quotes are checked.-> condn4
  if *any of the above condition(condn) is/are satisfied* **then**
    label tweet as sarcastic (1)
  else
    label tweet as non-sarcastic (0)
end

3.4.1 Data Collection and Pre-Processing

To collect the sarcastic tweet, the API was queried to return the tweets marked "'*#sarcastic*" which serves as a search indicator to detect the sarcastic tweets as they are subjective to a particular person and often elusive. Data was cleaned to remove noises following the steps in Algorithm 1.Finally, hashtag indicating sarcasm was removed and the data set was manually annotated (sarcastic tweets getting a value 1 and non-sarcastic having a value 0).

3.4.2 Sarcasm detection

To classify the tweets Convolution Neural Network algorithm [61] was used as shown in Fig.5.

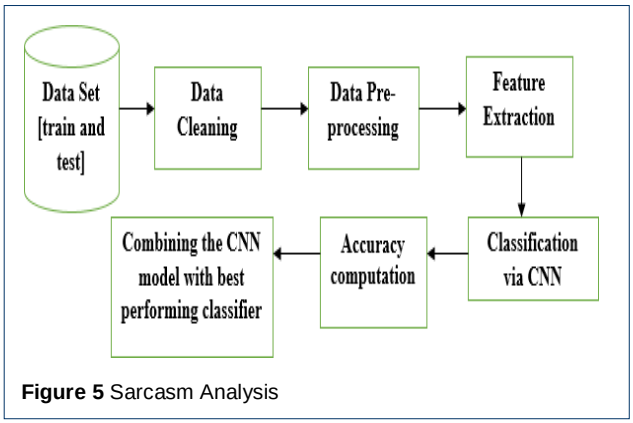

Figure 5 Sarcasm Analysis

3.5 Homophily detection

Unlike direct tweets, here the contacts and the com- munication elements of the users were examined, to deduce the chosen political party from similar users. The connections analyzed in our work are described below [62].

- *Following/Follower:* It is assumed that people take interest in accounts of personalities with whom they share certain similarity or whom they like. The idea here is to accumulate the fol- lower/following group of a particular user, so as to infer his political ideology from them. We checked if majority of a particular user's friends follow "Democrats" or "Republicans" and inferred ac- cordingly.
- *Mention*: The mention similarity checked how often a particular party's handle or prominent leader's profile are mentioned by a particular user using '@' in a positive sense. It also analyses how often a particular person's handle is mentioned in the tweets of user, to access their friendship (Fur- ther inference of the user's political choice is done using that of his aide).
- *Retweet* : Users retweeting a particular person's tweet , shows their interest, which can be garnered as a way to find similarity.
- *Hashtag Similarity* : Use of common hashtags trending with respect to a particular party wass scrutinized to infer the sentiments towards these by a particular user.

3.5.1 Classification

Using the above signals 2 groups per signal is generated named as *'Trump1'/'Hillary1','Trump2'/'Hillary2', 'Trump3'/'Hillary3'and'Trump4'/'Hillary4'*. Class was

calculated for each of the signal and final class was said to be the one which the majority signals outputs. Fig. 6 embellishes the entire process. Following classifiers were used to infer the political party via the combined signals.

1 Multinomial Na¨ıveBayes (MNB) [54]
2 Decision Tree (DT) [63]
3 Random Forest(RF) [55] [63]
4 Support Vector Machine (SVM) [56]
5 Linear Support Vector Classification (SVC) [64]
6 Logistic Regression (LR) [65]
7 Majority Voting of above classifiers (MNB,DT,RF,SVM,LR and Linear SVC) as explained in Algorithm 6 [66]
8 k-Nearest Neighbour with k=3 [67]

Algorithm 6 Majority Voting of the classifiers

Input : Signals,user j **Output:** political party
$ds_j \leftarrow 0$ $rs_j \leftarrow 0$
foreach *base classifier* c_i **do**
  if c_i *predicts democrat* **then**
    $ds_j \leftarrow ds_j + 1$
  else
    $rs_j \leftarrow rs_j + 1$
end
if $rs_j > ds_j$ **then**
  politicalparty = "REPUBLICAN"
else
  politicalparty = "DEMOCRATIC"
return political party

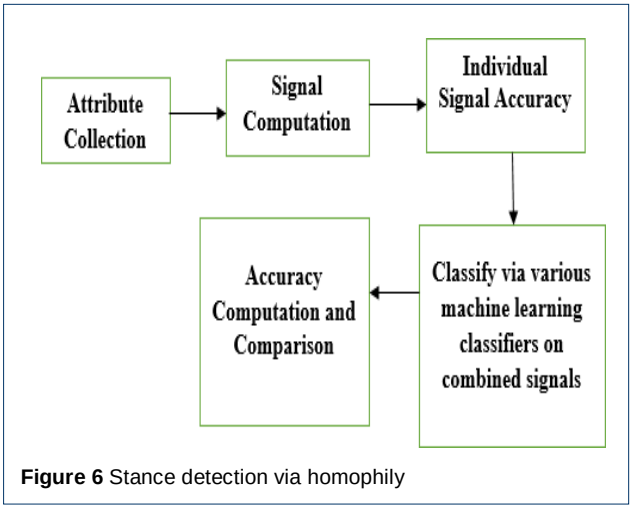

Figure 6 Stance detection via homophily

4 Result Analysis

4.1 Analysis of Tweets

Table 1 and Fig.7 exhibit the results observed when computing single tweet accuracy.

Table 1 Results for political class classification via tweets

Classifier	Accuracy (in percent)
Multinomial Naive Bayes	79.75
Random Forest Classifier	84.32
SVM	85.47
Bi-directional Long Short-term Memory (LSTM)	**87.5**
Proposed Ensemble Model	**88.61**

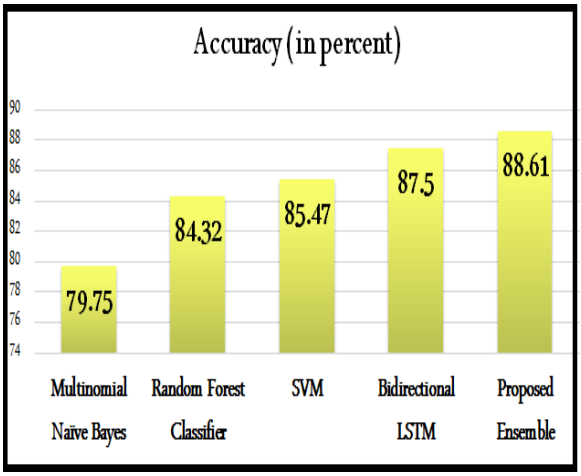

Figure 7 Results for political class classification via tweets

Sarcasm Detection with CNN gave an accuracy of 80%. Choice of CNN over RNN to detect sarcasm was done for the following reasons:

- CNN is faster than RNN.
- CNN is better at detecting emotional linguistics like anger, abuse etc. which is vital in detecting sarcasm.

When the model was annexed to the proposed Ensemble model it brought down the accuracy to 80.08%. The reason for this could be one of the following:

- The sarcasm detection model was trained on a set of tweets which were not political in nature. The reason behind this is that there was no available dataset where political tweets were labeled as sarcastic/not sarcastic. Due to this, there is a high probability that there were a large number of words which the model encountered for the first time while attempting to predict the political tweets, and hence didnot perform as well as it should have.
- The accuracy of the sarcasm detection model was only 80%.It is very difficult to increase this accuracy with the given constraints in terms of available dataset and subjective nature of sarcasm.

When the Ensemble model was applied to all the collected political tweets of a single user, the accuracy

rose to 97%.Here every tweet pertaining to politics was analysed and inferred class (Democratic/Republican) count was calculated.Final class was assigned to the one having higher class count.

4.2 Analysis of Metadata

The signals can be mathematically presented in the following manner [62].

$$Follow/Following\ S1 = n + k \quad (1)$$

where n is the total number of user $_i$'s followers and k is the total number of users user$_i$ is following.

Mention

$$S2 = \sum_{n=1}^{q} \frac{tthreadmen(n, u_i, u_j)}{tthreadtot(n, u_i)} \times \frac{1}{acctotthread(n, u_i)} \quad (2)$$

where

- *tthreadmen is a function that returns the number of u_i tweets in the communication thread n with u_j that mention the account u_j (can be a friend or a political party/its leader's handle).*
- *acctotthread is the total number of accounts in the tweets in thread n.*
- *tthreadtot is a function that returns the total number of tweets in the communication thread n.*
- *q is the total number of communication threads mentioning both u_i and u_j.*

$$Retweets\ S3 = noofTwtsreTweetd(u_i, u_j) \quad (3)$$

where the function noofTwtsreTweetd returns total number of u_j's (can be a friend or a political party/its leader's handle) tweets that u_i retweeted.

$$Hashtags\ S4 = \sum_{n=1}^{q} \frac{1}{1 + HashFunc(u_i, u_j, H_n)} \quad (4)$$

$$HashFunc(u_i, u_j, H_n) = |P(u_i, H_n) - P(u_j, H_n)| + |N(u_i, H_n) - N(u_j, H_n)| + |NU(u_i, H_n) - NU(u_j, H_n)|$$

where

- *function P takes userid and hashtag as input and returns total positive tweets tweeted on the hashtag by the user.*
- *function N takes userid and hashtag as input and returns total negative tweets tweeted on the hashtag by the user.*
- *function NU takes userid and hashtag as input and returns total neutral tweets tweeted on the hashtag by the user.*
- *q is the total number of hashtags tweeted by both u_i and u_j*

Table 2 and Fig.8 exhibit the results observed during individual signal computation.

Table 2 Results for individual signal accuracy

Signal	Accuracy (in percent)
Follower/Following	82.11
Mention	11.58
Retweet	26.70
Hashtag Similarity	50.12

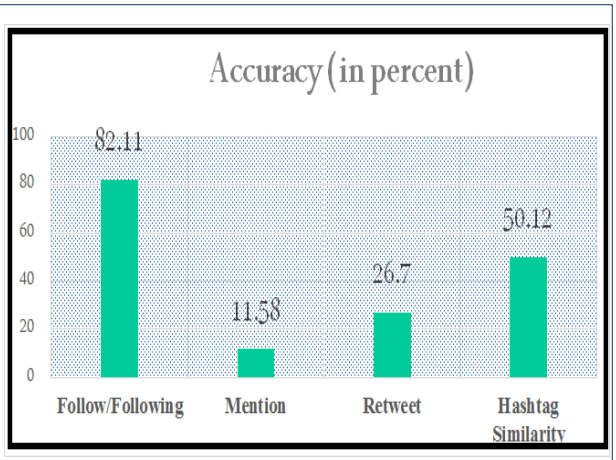

Figure 8 Results for individual signal accuracy

Results for prediction using combined signals are shown in Table 3 and Fig. 9.

Table 3 Results for political class classification via all signals

Classifier	Accuracy (in percent)
Multinomial Naive Bayes	85
Decision Tree	83.11
Logistic Regression	86.45
Random Forest Classifier	87.32
Linear SVC	89.11
SVM	84.33
Majority Voting of above classifiers	**90**
k-NN	**93**

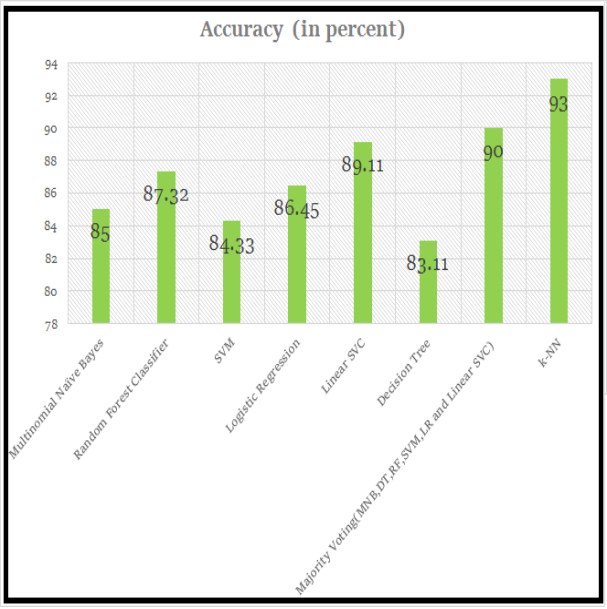

Figure 9 Results for political class classification via all signals

5 Conclusion and Future Work

Twitter, a habitual podium nowadays, is widely used by many subscribers to share their moments and their stand on various issues oscillating from current affairs to mental health etc. via texts, support (retweets, likes, and mentions), multimedia, emoji and groups. Existing work emphasized only on the exploration of the tweets of the users to predict the party they vouch for. However, it was found to be ineffective in the situations where the users don't clarify their stand towards a particular political party explicitly. A single parameter like follow/following is not a viable measure to cluster similar users together, as their stance on all topics need not be the same. Hence, to address the above mentioned research gaps, the proposed methodology incorporated a dual process, wherein the first section analysed the prominent political tweets of a particular user, so as to infer which political party he prefers. This was supplemented with effectively categorizing the sarcastic tweets and the emoji used. Additionally, using the user's network (follower/following, retweets, mentions and hashtags), similar users were clustered together, following which his stance was inferred from his peers.

Although both the methods gave a good accuracy score, they can be enhanced by including additional signals such as profile type similarity, likes, topography, geography etc. subjective to the application construed on. Content analysis of the bio can also assist in grouping people who are alike. A notable future work would be detecting the strength of user connections (uni/bidirectional and time period-length of interactions) in a specific context (ex. political) to improve the homophily detection . Sarcasm and emoticon analysis can be improved upon by acquiring suitable datasets (containing more suitable Twitter dialects).Inferring information from non-English words (mostly in Indian context), URLs, images and neutral tweets can be a new domain of research. Noteworthy domains where this research can be extended are e-commerce, marketing analysis, finding online trolls etc.

Author details

[1]Department of Information and Communication Technology, Manipal Institute of Technology,Manipal Academy of Higher Education, Manipal, India. [2]Department of Computer Science and Engineering, Manipal Institute of Technology,Manipal Academy of Higher Education, Manipal, India.

References

1. Y. Lin, "10 twitter statistics every marketer should know in 2020 [infographic]," Jul 2020. [Online]. Available: https://www.oberlo.in/blog/twitter-statistics.
2. E. Raad and R. Chbeir, "Privacy in Online Social Networks," in Security and Privacy Preserving in Social Networks. Springer-Verlag Wien, 2013, pp. 3–45. [Online]. Available: https://hal.archives-ouvertes.fr/hal-00975998.
3. J. Gehrke, E. Lui, and R. Pass, "Towards privacy for social networks: A zero-knowledge based definition of privacy," in Theory of Cryptography, Y. Ishai, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 432–449.
4. J. H. Kietzmann, K. Hermkens, I. P. McCarthy, and B. S. Silvestre, "Social media? get serious! understanding the functional building blocks of social media," Business Horizons, vol. 54, no. 3, pp. 241 – 251, 2011, sPECIAL ISSUE: SOCIAL MEDIA. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0007681311000061.
5. D. M. E.-D. M. Hussein, "A survey on sentiment analysis challenges," Journal of King Saud University - Engineering Sciences, vol. 30, no. 4, pp. 330 – 338, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1018363916300071
6. B. Bansal and S. Srivastava, "On predicting elections with hybrid topic based sentiment analysis of tweets," Procedia Computer Science, vol. 135, pp. 346 – 353, 2018, the 3rd International Conference on Computer Science and Computational Intelligence (ICCSCI 2018) : Empowering Smart Technology in Digital Era for a Better Life. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050918314728
7. M. Devika, C. Sunitha, and A. Ganesh, "Sentiment analysis: A comparative study on different approaches," Procedia Computer Science, vol. 87, pp. 44 – 49, 2016, fourth International Conference on Recent Trends in Computer Science Engineering (ICRTCSE 2016). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187705091630463X
8. J. A. Caetano, H. S. Lima, M. F. Santos, and H. T. Marques-Neto, "Using sentiment analysis to define twitter political users' classes and their homophily during the 2016 american presidential election," Journal of Internet Services and Applications, vol. 9, no. 1, 2018.
9. S. Al-Azani and E.-S. M. El-Alfy, "Using word embedding and ensemble learning for highly imbalanced data sentiment analysis in short arabic text," Procedia Computer Science, vol. 109, pp. 359 – 366, 2017, 8th International Conference on Ambient Systems, Networks and Technologies, ANT-2017 and the 7th International Conference on Sustainable Energy Information Technology, SEIT 2017, 16-19 May 2017, Madeira, Portugal. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050917310347

10. S. Seo, C. Kim, H. Kim, K. Mo, and P. Kang, "Comparative study of deep learning-based sentiment classification," IEEE Access, vol. 8, pp. 6861–6875, 2020.
11. N. Oliveira, J. Costa, C. Silva, and B. Ribeiro, "Retweet predictive model for predicting the popularity of tweets," Proceedings of the Tenth International Conference on Soft Computing and Pattern Recognition (SoCPaR 2018) Advances in Intelligent Systems and Computing, p. 185–193, 2019.
12. P. Nesi, G. Pantaleo, I. Paoli, and I. Zaza, "Assessing the retweet proneness of tweets: predictive models for retweeting," Multimedia Tools and Applications, vol. 77, no. 20, p. 26371–26396, 2018.
13. S. Sharma and N. P. Shetty, "Determining the Popularity of Political Parties Using Twitter Sentiment Analysis," Advances in Intelligent Systems and Computing Information and Decision Sciences, pp. 21–29, 2018.
14. W. Budiharto and M. Meiliana, "Prediction and analysis of indonesia presidential election from twitter using sentiment analysis," Journal of Big Data, vol. 5, no. 1, 2018.
15. I. E. Alaoui, Y. Gahi, R. Messoussi, Y. Chaabi, A. Todoskoff, and A. Kobi, "A novel adaptable approach for sentiment analysis on big social data," Journal of Big Data, vol. 5, no. 1, 2018.
16. D. Xue, L. Wu, Z. Hong, S. Guo, L. Gao, Z. Wu, X. Zhong, and J. Sun, "Deep learningbased personality recognition from text posts of online social networks," Applied Intelligence, vol. 48, no. 11, p. 4232–4246, 2018.
17. S. M. Nagarajan and U. D. Gandhi, "Classifying streaming of twitter data based on sentiment analysis using hybridization," Neural Computing and Applications, vol. 31, no. 5, p. 1425–1433, 2018.
18. A. Kumar, A. Ekbal, D. Kawahra, and S. Kurohashi, "Emotion helps sentiment: A multitask model for sentiment and emotion analysis," in 2019 International Joint Conference on Neural Networks (IJCNN), 2019, pp. 1–8.
19. R. Wang, D. Zhou, M. Jiang, J. Si, and Y. Yang, "A survey on opinion mining: From stance to product aspect," IEEE Access, vol. 7, pp. 41 101–41 124, 2019.
20. H. Suresh and G. R. S, "An innovative and efficient method for twitter sentiment analysis," International Journal of Data Mining, Modelling and Management, vol. 11, no. 1, p. 1, 2019.
21. A. Sharma and U. Ghose, "Sentimental analysis of twitter data with respect to general elections in india," Procedia Computer Science, vol. 173, pp. 325 – 334, 2020, international Conference on Smart Sustainable Intelligent Computing and Applications under ICITETM2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050920315428
22. M. Z. Ansari, M. Aziz, M. Siddiqui, H. Mehra, and K. Singh, "Analysis of political sentiment orientations on twitter," Procedia Computer Science, vol. 167, pp. 1821 – 1828, 2020, international Conference on Computational Intelligence and Data Science. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050920306669
23. G. Rossetti, L. Pappalardo, R. Kikas, D. Pedreschi, F. Giannotti, and M. Dumas, "Homophilic network decomposition: a community-centric analysis of online social services," Social Network Analysis and Mining, vol. 6, no. 1, 2016.
24. K. Darwish, W. Magdy, and T. Zanouda, "Improved stance prediction in a user similarity feature space," in Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, ser. ASONAM '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 145–148. [Online]. Available: https://doi.org/10.1145/3110025.3110112.
25. J. Du, R. Xu, Y. He, and L. Gui, "Stance classification with target-specific neural attention networks," in Proceedings of the 26th International Joint Conference on Artificial Intelligence, ser. IJCAI'17. AAAI Press, 2017, p. 3988–3994.
26. Gayathri Rajendran, Bhadrachalam Chitturi, Prabaharan Poornachandran, "'Stance-In-Depth Deep Neural Approach to Stance Classification"', Procedia Computer Science,Volume 132,2018,Pages 1646-1653.
27. L. Poddar, W. Hsu, M. L. Lee, and S. Subramaniyam, "Predicting stances in twitter conversations for detecting veracity of rumors: A neural approach," in 2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI), 2018, pp. 65–72.
28. M. Barone and M. Coscia, "Birds of a feather scam together: Trustworthiness homophily in a business network," Social Networks, vol. 54, pp. 228 – 237, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378873317300370.
29. M. Lai, M. Tambuscio, V. Patti, G. Ruffo, and P. Rosso, "Stance polarity in political debates: A diachronic perspective of network homophily and conversations on twitter," Data Knowledge Engineering, vol. 124, p. 101738, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169023X19300187.
30. K. Sailunaz and R. Alhajj, "Emotion and sentiment analysis from twitter text," Journal of Computational Science, vol. 36, p. 101003, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877750318311037.
31. A. Aldayel and W. Magdy, "Your stance is exposed! analysing possible factors for stance detection on social media," Proceedings of the ACM on Human-Computer Interaction, vol. 3, pp. 1 – 20, 2019.
32. Hamdi, Tarek, et al., "A Hybrid Approach for Fake News Detection in Twitter Based on User Features and Graph Embedding.", Distributed Computing and Internet Technology, Lecture Notes in Computer Science, 2019, pp. 266–280.
33. K. Darwish, P. Stefanov, M. J. Aupetit, and P. Nakov, "Unsupervised user stance detection on twitter," in ICWSM, 2020.
34. H. Fede, I. Herrera, S. M. M. Seyednezhad, and R. Menezes, "Representing Emoji Usage Using Directed Networks: A Twitter Case Study," Studies in Computational Intelligence Complex Networks & Their Applications VI, pp. 829–842, 2017.
35. Y. Chen, J. Yuan, Q. You, and J. Luo, "Twitter sentiment analysis via bi-sense emoji embedding and attention-based lstm," 2018 ACM Multimedia Conference on Multimedia Conference - MM 18, 2018.
36. J. Subramanian, V. Sridharan, K. Shu, and H. Liu, "Exploiting Emojis for Sarcasm Detection," Social, Cultural, and Behavioral Modeling Lecture Notes in Computer Science, pp. 70–80, 2019.
37. D. qin Li, R. Rzepka, M. Ptaszynski, and K. Araki, "Emoji-aware attentionbased bi-directional gru network model for chinese sentiment analysis," in LaCATODA/ BtG@IJCAI, 2019.
38. Y. Cai, H. Cai, and X. Wan, "Multi-modal sarcasm detection in twitter with hierarchical fusion model," in ACL, 2019.
39. R. A. Potamias, G. Siolas, and A.- G. Stafylopatis, "A transformer-based approach to irony and sarcasm detection," Neural Computing and Applications, 2020.
40. S. Khotijah, J. Tirtawangsa, and 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, ser. IAIT2020. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: https://doi.org/10.1145/3406601.3406624.
41. K. Sundararajan and A. Palanisamy, "Multi-rule based ensemble feature selection model for sarcasm type detection in twitter," Computational Intelligence and Neuroscience, vol. 2020, p. 1–17, 2020.
42. K. Pastor, "Democrat vs. republican tweets," May 2018. [Online]. Available: https://www.kaggle.com/kapastor/democratvsrepublicantweets.
43. Shashanksai, "Text preprocessing using python," Sep 2018. [Online]. Available: https://www.kaggle.com/shashanksai/text-preprocessing-using-python.
44. D. Monsters, "Text preprocessing in python: Steps, tools, and examples," Oct 2018. [Online]. Available: https://medium.com/@datamonsters/text-preprocessing-in-python-steps-tools-and-examples-bf025f872908.
45. "(tutorial) text analytics for beginners using nltk." [Online]. Available: https: //www.datacamp.com/community/tutorials/text-analytics-beginners-nltk.
46. "Emojipedia." [Online]. Available: https://emojipedia.org/.
47. S. Lekach, "The real meaning of all those emoji in twitter handles," Jun 2017. [Online]. Available: https://mashable.com/2017/06/03/emoji-twitter-handles-meanings/.
48. J. Zote, "130 most important social media acronyms and slang you should know," May 2020. [Online]. Available: https://sproutsocial.com/insights/social-media-acronyms/.

49. V. Beal, “Twitter dictionary: A guide to understanding twitter lingo.” [Online]. Available: https://www.webopedia.com/quick ref/Twitter Dictionary Guide.asp.
50. B. Billal, A. Fonseca, and F. Sadat, “Efficient natural language pre-processing for analyzing large data sets,” in 2016 IEEE International Conference on Big Data (Big Data), 2016, pp. 3864–3871.
51. A. Deniz and H. E. Kiziloz, “Effects of various preprocessing techniques to turkish text categorization using n-gram features,” in 2017 International Conference on Computer Science and Engineering (UBMK), 2017, pp. 655–660.
52. J. Brownlee, “How to encode text data for machine learning with scikitlearn,” Jun 2020. [Online]. Available: https://machinelearningmastery.com/ prepare-text-data-machine-learning-scikit-learn/
53. C. Maklin, “Tf idf: Tfidf python example,”Jul 2019. [Online]. Available: https://towardsdatascience.com/natural-language-processing-feature-engineering-using-tf-idf-e8b9d00e7e76
54. S. Prabhakaran, “How naive bayes algorithm works? (with example and full code): Ml,” Nov 2018. [Online]. Available: https://www.machinelearningplus.com/ predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/
55. A. Cuzzocrea, S. L. Francis, and M. M. Gaber, “An information-theoretic approach for setting the optimal number of decision trees in random forests,” 2013 IEEE International Conference on Systems, Man, and Cybernetics, 2013.
56. Ml - support vector machine(svm).” [Online]. Available: https://www.tutorialspoint.com/machine learning with python/machine learning with python classification algorithms support vector machine.htm.
57. “Understanding lstm networks.” [Online]. Available: https://colah.github.io/posts/ 2015-08-Understanding-LSTMs/.
58. M. Phi, “Illustrated guide to lstm’s and gru’s: A step by step explanation,” Jun 2020. [Online]. Available: https://towardsdatascience.com/ illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21.
59. T. Liu, T. Wu, M. Wang, M. Fu, J. Kang, and H. Zhang, “Recurrent neural networks based on lstm for predicting geomagnetic field,” in 2018 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES), 2018, pp. 1–5.
60. Ankit, Nabizath Saleena,An Ensemble Classification System for Twitter Sentiment Analysis, Procedia Computer Science,Volume 132,2018,Pages 937-946,ISSN 1877-0509,Available: https://doi.org/10.1016/j.procs.2018.05.109.
61. S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” in 2017 International Conference on Engineering and Technology (ICET), 2017, pp. 1–6.
62. AlMahmoud, H., AlKhalifa, S. TSim: a system for discovering similar users on Twitter. J Big Data 5, 39 (2018).Available: https://doi.org/10.1186/s40537-018-0147-2.
63. A. Munther, A. Alalousi, S. Nizam, R. R. Othman and M. Anbar, ”Network traffic classification — A comparative study of two common decision tree methods: C4.5 and Random forest,” 2014 2nd International Conference on Electronic Design (ICED), Penang, 2014, pp. 210-214, doi: 10.1109/ICED.2014.7015800.
64. Q. Li, Y. Fu, X. Zhou and Y. Xu, ”The Investigation and Application of SVC and SVR in Handling Missing Values,” 2009 First International Conference on Information Science and Engineering, Nanjing, 2009, pp. 1002-1005, doi: 10.1109/ICISE.2009.1226.
65. L. Liu, ”Research on Logistic Regression Algorithm of Breast Cancer Diagnose Data by Machine Learning,” 2018 International Conference on Robots & Intelligent System (ICRIS), Changsha, 2018, pp. 157-160, doi: 10.1109/ICRIS.2018.00049.
66. R. Atallah and A. Al-Mousa, ”Heart Disease Detection Using Machine Learning Majority Voting Ensemble Method,” 2019 2nd International Conference on new Trends in Computing Sciences (ICTCS), Amman, Jordan, 2019, pp. 1-6, doi: 10.1109/ICTCS.2019.8923053.
67. S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, “Efficient knn classification with different numbers of nearest neighbors,” IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 5, pp. 1774–1785, 2018.

Abbreviations

- OSN -Online Social Network
- CNN - Convolution Neural Network
- RNN - Recurrent Neural Network
- LSTM -Long Short-term Memory
- k-NN - k Nearest Neighbour
- SVM -Support Vector Machine
- DBSCAN -Density-based spatial clustering of applications
- GRU- Grated Recurrent Unit
- TFIDF- Term frequency-inverse document frequency