International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 /doi:10.14445/22315381/IJETT-V69I1P213

Volume 69 Issue 1, 85-91, January 2021

Social Media Behavioural Analysis With Document Tree-Based Rule Mining and Document Clustering

S. Geetha[1], Dr. R. Kaniezhil[2#]

Research Scholar, Department of Computer Science, Periyar University, Salem, Tamil Nadu, India
Principal, MIT College of Arts & Science for Women, Musiri, Tamil Nadu, India

[1]geeth_s20@yahoo.com,[2]kaniezhil@yahoo.co.in

Abstract *- Twitter in Social media has become an important part of regular lives. This media provides a list of trending real-time topics where most information is hard to comprehend, making it imperative to classify for finding useful information. A large database with real-time information is generated on Twitter. Twitter tweets are a storehouse of text and can reflect human emotions and feelings. Hidden information found in this data can be used for multiple purposes. However, the results depend on choosing a proper feature set. Human biological, pharmacological, and experiential factors influence their behavior. Behavior Analysis (BA) is analyzing individual behavior. BA can be used to filter useful information from tweets in healthcare and business applications. This paper proposes an analysis of human behavior using Twitter data with the proposed DRDC algorithm. The proposed algorithm uses a multitude of techniques in its pre-processing, feature selection, and classification of tweets. Further, the algorithm's accuracy is checked using the factors of precision and recall times.*

Keywords: *Behavioral Analysis, Social Media Data Sets, Decision Trees, Document Clustering, Stemming, Pre-processing, DRDC*

I. INTRODUCTION

Humans are generally prejudiced and partial in their views as they are inclined towards others' opinions, which can be found in almost all types of civilizations. SM (Social Media) platforms have become an important medium for public conversations in this communication world. The growth of the internet and SM have enabled people to communicate freely. SM has information on people's interactions, opinions, and expressions about happenings in the world. Thus, SM users share their thought or opinions on any subject that can be viewed [1] [2]. Internet/smartphone explosions have made it possible to opine even from remote areas in the world. SNSs (Social Networking Sites) like Facebook or Twitter or Instagram has millions of users who comment on world happenings. Over 4 billion people are active online, accounting for 55% of the world's population. Digital Transformations are moving rapidly, generating interest in people to be in touch with each other. SM platforms carry information on subjects expressed in text form and published on blogs or comments, or web reviews. Figure 1 displays Digital users in the world as of July 2020.

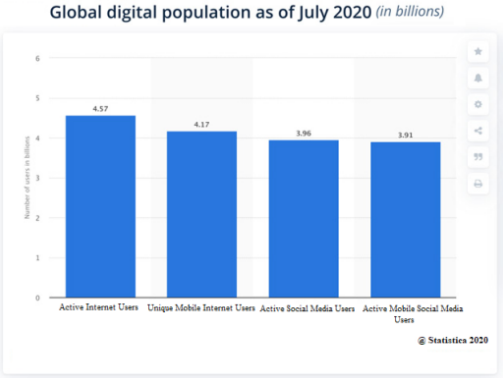

Fig. 1 : **Worldwide Digital Usage as of July 2020 [3]**

SNSs have become a gold mine for analyzing public behavior or judging their moods. People's behavior and opinions on events can be assessed by organizations from expressions of users from SNSs. Thus, BA (Behaviour Analysis) becomes an important tool in this application area [4]. BA as a study gains its significance in research as it can decipher public trends or judge attitudes, opinions, and trends from publicly available data. A detailed analysis in BAs can project human desires or love or disgust or likes or dislikes. These publicly displayed emotions can be classified positively or negatively based on their contexts [5]. Twitter is a public forum where millions participate and express their innate feelings on a variety of topics.

Moreover, 70% of Twitter users participate in text conversations (Tweets) daily, creating a warehouse of behavioral information. This warehouse also opens up the gates for research in DM (Data Mining), assessing trends, behavior, and opinions by mining information. This paper contributes towards DM of sentiments by assessing users' behavior in Twitter by proposing a novel technique called DRDC, which uses a combination of rule mining and document clustering. The next section of this paper presents a review of the literature followed by the DRDC methodology. The implementation results are presented in the fourth section, followed by the paper's conclusion in section five.

II. RELATED WORK ON BA

BA in research is studying user behavior as they reflect public opinions. BAs have been applied in health care, organizational monitoring, and business promotions. BA research can also address philosophical, historical, theoretical, and methodological issues faced while analyzing data. Applied behavior analysts have addressed behaviors associated with autism, developmental disorders while helping healthcare treatment. BA can identify changes even in huge volumes of behavioral data using NLP (National Language Processing). BA has been used to extract opinions from people's behaviors using NLP based on specified contexts [6]. Though BA has achieved expected targets, its executions or selections have not been an easy task as opinions are expressed in short forms or full sentences, or ironical forms. ML (Machine Learning) Techniques have been greatly used to assess public opinions on subjects by BA. BA can also be related to OM (Opinion Mining), RA (Effect Analysis), and RM (Review Mining) [7]. Though SA (Sentiment Analysis) is similar to BA in many aspects, intricate differences in looking ate their characteristics differentiate them [8]. Other similar methods only extract inferences from pulic opinions, while BA involves predicting the future by examining underlying unexpected attitudes in haviours expressed in the text. Thus, BA has found global use like predicting election results [9] or stock market trends [10] in advance. Organizations have been attempting to exploit BA techniques to their maximum advantage by studying their customer behavior or corresponding market bases [11]. Most documents analyzed in similar kinds of operations fail to analyze underlying conditions, while BA techniques exemplify these texts or documents' nature and origin. BA techniques are successful as they classify their document data as subjective or objective before any analysis is done on the documents. Twitter amongst SNS is one of the most visited, and its information can be used in BA, which is similar to sentiment analysis [12]. BA has a unique problem in the micro-blogging domain analysis works on well-formed data [12] [14] [15] [16] as informal language is used in tweets. Researches have investigated BA techniques, including parts-of-speech features and several areas like an automatic collection of training data from tweepy API. BA can be Lexicon Based (LB) and Machine Learning-Based (MLB). DB techniques use the predefined dictionary for Behavior classifications, while MLB systems use data from required specific domains.

Imbalances in Text classes or Linguistic dissimilarities are solved by bootstrapping [17]. A combination of SVM and clustering was used to classify Twitter data in [18], which demonstrated societal attitude and emotions in tweets. Twitter sentiments were manipulated using hashtags to classify tweets as likes and dislikes in [19]. ANN (Artificial Neural Networks) based content ranking was proposed in [20] to identify user's content polarity. A data collection method framework was used for analyzing sentiments in [21], while [22] analyzed stock market tweets using NN (Neural Networks) and SVM (Support Vector Machines). NB (Naive Bayes) was used to identify emotions from Facebook pages in [23]. Existing BA techniques are detailed in Table I.

Table 1 : Current Methodologies of BA

	LB	MLB
Approach	Dictionary-based Classification	Probabilistic classification where learning can be Unsupervised or Supervised.
Advantage	Minimized computational requirements	Domain-specific customization
Limitation	Deficiency in the classification of required spheres.	Class imbalances or linguistic dissimilarities can arise as a problem.

This paper proposes a BA technique called **Decision tree-based Rule mining and Document Clustering** (DRDC), which uses rule mining, decision trees, and clustering. Before applying BA techniques, the data in consideration transform Every word in a title must be capitalized except for short minor words such as a, an, and, as, at, by, for, from, if, in, into, on, or, of, the, to, with.

III. DRDC ARCHITECTURE

Twitter being a popular Social Networking site amongst organizations and news providers, the proposed technique uses Twitter tweets for in BA. DRDC uses three distinctive steps in its analysis of tweets. The dataset is first pre-processed with Stop word selection and stemming. The pre-processed data then undergoes a feature selection process using rule elimination and Levenshtein Algorithm. This output is then classified. Though stemming and stopwords removal is a dimensionality reduction function, DRDC uses these steps in its pre-processing to clean and reduce data. Stemming reduces dimensionality by identifying the root of a word. When it is used with a dictionary, it can also correct misspelt words. In Feature Selection, the DRDC processes of grammatical rules of negation can help improve sentiment or behavior classification accuracy. The processed rules represent a subset of grammatical rules for the processing of negation in the language, i.e., the most commonly encountered rules in tweets. This is followed by the use of the Levenshtein Algorithm in DRDC to remove redundancy in Tweets. The selected features are then processed by DRDC for classification, where Decision Trees(DT), rule mining, and Document Clustering (DC) are used. DT

constructs its decisions in a top-down approach. It finally culminates in providing tuples belonging to the same class. DRDC then uses Association Rule Mining (ARM) on the DT output. DC is generally used in IR and the final step in DRDC classifications. DRDC architecture is depicted in Figure 1.

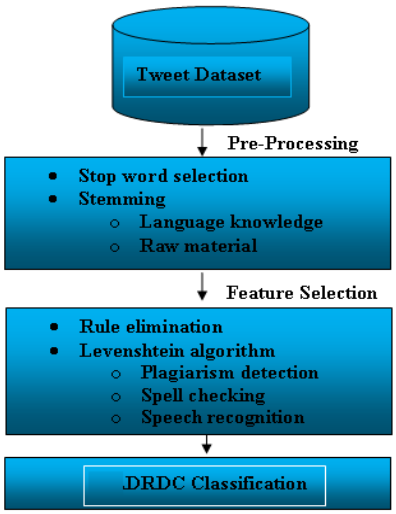

Fig. 2 : DRDC Architecture

A. DRDC Preprocessing

Stop word selection is the first process executed by DRDC in its pre-processing of tweets. Stop words are screened due to their low discrimination power in communications. Example being "the", "is", "at", "which", and "on". Tweet data stop words are filtered before BA as they can influence the final performance of BA. The study in [24] examined used six tweet datasets for gauging the impact of stop words on effective BA. DRDC identifies Stop Words for removal, and only the filtered information is processed further. Figure 2 depicts Tweets with Stop Words.

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

Fig. 3 : Tweet Stop Words

Tweet words that have similar meanings are stemmed, or the root/stem of a word is found. Stemming is neutralizing suffixes and prefixes in a word. DRDC uses a dictionary as its raw material for looking up language meanings. The next section details the featire selection of this final pre-processed output. Figure 3 depicts the stem Consult and its corresponding variances.

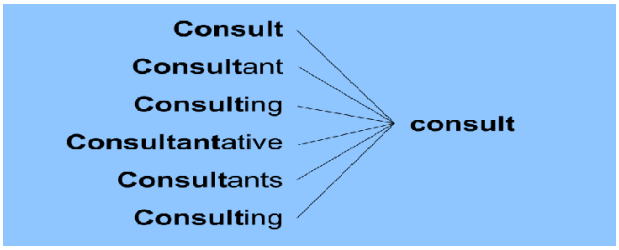

Fig. 4 : **Stem of the word Consult**

B. DRDC Feature Selection

Grammatical rules influence polarity and can cause a significant improvement in BA. Advanced systems take into account both POS tags, apply the rules of negation and detect irony. Although the determination of polarity, as one of BA's tasks, is an area that has been studied to some extent, some linguistically specific aspects (negation, irony, metaphor) still pose challenges and areas where improvements are expected. The study in [25] identified negations and created rules to improve BA output. The method predicted behavior as an unsupervised method that calculated the polarity and intensity of the words and phrases. The study showed a positive correlation between the polarity determined by the system and participant assigned words. Feature selection is “ultra” are not independent words; they should be joined to the words they modify, usually without a hyphen. There is no period after the “et” in the Latin abbreviation “et al.” (it is also italicized). The abbreviation, “i.e.,” means “that is,” and the abbreviation “e.g.,” means “for example” (these abbreviations are not italicized).

DRDC is followed by using elimination or negation rules and the use of the Levenshtein Algorithm. The rules of negation followed in DRDC are:

- Treating negation like Not only .... but/instead – the word in front of the word “only” is omitted.
- Treating negation like (Not .... but/instead) – as the negation's scope ends with the word “but/ instead,” it is omitted.
- Treating negation like “(“Is it not nice?”) - is neutralized as the negation does not affect the polarity of sentiments.
- In the presence of an intensifier after a negated word, the next part of the sentence is negated or filtered.
- Negative quantifiers increase the intensity of the negation and hence are not considered in selections.

The rules were applied in the order they are given, while Levenshtein distance between words was calculated using Equation (1)

$$\mathrm{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \mathrm{lev}_{a,b}(i-1,j) + 1 \\ \mathrm{lev}_{a,b}(i,j-1) + 1 \\ \mathrm{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

……..(1)

where 1(ai≠bi), the indicator function equals zero or one when they are equal, and lev $_{a,\ b}$(i,j) is the distance between the 1st i characters of a and 1st j characters of b. Levenshtein algorithm is designed for measuring edit distance [26]. The term "edit distance*"* is used for calculating the difference between two strings or operations required on a word to get transformed [27]. The operations of the Levenshtein algorithm are performed on a single symbol or a single character, and they consist of insertion, deletion, and substitution. Each operation is treated as a single edit [28]. If in a given query, "*csp*" is a non-English word, the Levenshtein algorithm needs to perform one substitution to transform the world into "*cup,*" an accepted English word [29]. Hence, LA's processing time will increase as it requires creating an array and filling up each cell with the word from a dictionary [30]. When used in generating candidates for spelling correction, the Levenshtein algorithm requires a million calculations for each incorrect word because most lexicons contain millions of words [31]. The Levenshtein algorithm computes the minimum no of changes required for changing one string to another, and its programming approach is shown in figure 4.

1. **A matrix is initialized measuring in the (m, n) cell the Levenshtein distance between the m-character prefix of one with the n-prefix of the other word.**
2. **The matrix can be filled from the upper left to the lower right corner.**
3. **Each jump horizontally or vertically corresponds to an insert or a delete, respectively.**
4. **The cost is normally set to 1 for each of the operations.**
5. **The diagonal jump can cost either one, if the two characters in the row and column do not match else 0, if they match. Each cell always minimizes the cost locally.**
6. **This way the number in the lower right corner is the Levenshtein distance between both words.**

Fig. 5 : **Levenshtein Programming Steps**

Each cell value in the Levenshtein array needs eight operations: compare (3), add (3) and assign (2). Such inefficiency motivates for Levenshtein algorithm improvement that reduces the operational process without affecting its accuracy. This algorithm is used by DRDC *for* Matching strings and checking Spellings. *Figure 5 depicts the* distance between the words "HONDA" and "HYUNDAI," 3.

		H	Y	U	N	D	A	I
	0	1	2	3	4	5	6	7
H	1	0	1	2	3	4	5	6
O	2	1	1	2	3	4	5	6
N	3	2	2	2	2	3	4	5
D	4	3	3	3	3	2	3	4
A	5	4	4	4	4	3	2	3

Fig. 6 : **Levenshtein Sample Output**

C. DRDC Classification

Decision Trees are used as the first step in DRDC classification as it is an excellent tool for choosing between several courses of action.DT inducts relevant features in the form of nodes (Leaves) for testing attributes/features. The branch which holds the leaf in a DT classification represents outcomes while the leaf represents the prediction of a class. The most fir attribute in DT is chosen based on the Information gain predicted using Equation (2)

$$\inf o(D) = -\sum_{i=1}^{m} p_i \log_2(p)$$

…….(2)

Where k is no of tweets, L is characters length, t is a single tweet, d is a distance, and m is a max tweet. DT's output is analyzed further using ARM for identifying data associations, correlations, and recurring patterns. ARM has an antecedent (if) and a consequence (then) found in the data items. ARM uses two main parameters, namely Support (Frequently relationships that appear in the data) and Confidence, which find the count of such discovered relationships to be true. Figure 6 depicts an example of ARM output.

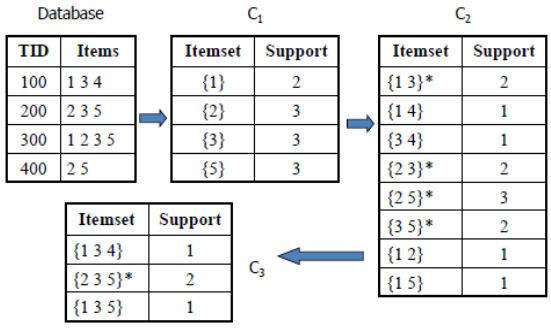

Fig. 7 : ARM Example

DC was used for investigating IR systems precision and recall results [32], while Clustering was used to organize documents [33] and return results of a query [34]. DC has also generated hierarchical clusters of documents [35], and hence DRDC uses DC for its efficiency. The algorithmic steps are listed below

DRDC Algorithmic Steps

Step 1: Set k to Character Length L
Set m or t lengths
If k = 0; exit
If m = 0; exit with value of k.
Create a matrix [0..m, 0..k]
Step 2: Initialize the first row and column to 0.
Step 3: Check each L for j (1..k) and t (j from 1..m).
Step 4: If L [j] =t [k] then cost = 0
Else cost = 1.
Step 5: Set d [J.k] to min. value = a.
The subsequent higher unit + 1=d [j-1,k] +1.
A add 1to the left of cell d [j,k-1] +1.
Add cost of diagonally up and left cells d [j-1,k-1] + cost.
Step 6: After iterating steps (3 to 6), find the distance J in cell d [k, m].
Where k – no of tweets, L – characters length, t – single tweet, d – distance, and m- max tweets

IV. RESULTS

This section describes stagewise experiments done on windows/ intel i5 CPU with 4 GB RAM in Python. Tweets were downloaded using Twitter API. Table 2 lists the records taken from training and positive and negative classification counts.

Table 2 Training/Test Data Details

Train Data	38000
Negative	19514
Positive	18486

Test Data	36832
Negative	19606
Positive	19226

A. DRDC Pre-processing

Pre-processing started with Stop words removal and was executed using a function. Figure 8 depicts the stop words list used by DRDC.

```
"i"         "me"        "my"          "myself"      "we"
"our"       "ours"      "ourselves"   "you"         "your"
"yours"     "yourself"  "yourselves"  "he"          "him"
"his"       "himself"   "she"         "her"         "hers"
"herself"   "it"        "its"         "itself"      "they"
"them"      "their"     "theirs"      "themselves"  "what"
"which"     "who"       "whom"        "this"        "that"
"these"     "those"     "am"          "is"          "are"
"was"       "were"      "be"          "been"        "being"
"have"      "has"       "had"         "having"      "do"
```

Fig. 8 : DRDC Stop Words List

After removing stop words based on the list Stemming, which usually concerns suffixes, the resulting stop words removed data. Figure 8 depicts DRDC sample Stemming Output.

	original_word	stemmed_words
0	connect	connect
1	connected	connect
2	connection	connect
3	connections	connect
4	connects	connect

Fig. 9 : **DRDC Sample Stemming Output**

DRDC feature selection starts with the application of negation rules followed by the use of Levenshtein distance. Figure 9 depicts the DRDC negation output.

DRDC Levenshtein Outputs

a	1
girl	2
smart	3
with	4
whom	5
I	6
Want	7
to	8
marry	9

Levenshtein Plagiarism Output

		b	c	a	k	s
		0	1	2	3	4	5
a		1	1	2	2	3	4
z		2	2	2	3	3	4

Levenshtein Spell Check Output

Fig. 10 : **DRDC Levenshtein plagiarism output**

B. DRDC Classification

DT is used as a first step as they are not affected by noisy data while learning on even disjunctive expressions. DT produces an effective structure with options for investigation and probable outcomes from options. DRDC classification starts with DT, followed by Rule Mining, which identifies frequent items (words) before using document clustering for its final Output. DRDC was evaluated with NB and SVM. Table 3 lists Comparative accuracies of Algorithms in Pre-processing.

Table 3 : **Comparative accuracies of Algorithms in Pre-processing**

Dataset Items	NB	SVM	DRDC
10	0.424417	0.457728	0.477828
50	0.56615	0.573375	0.583425
100	0.576708	0.592676	0.602726
500	0.63228	0.639318	0.649368
1000	0.656804	0.663723	0.673773
5000	0.663874	0.679603	0.689653
10000	0.700787	0.713355	0.723405
20000	0.725049	0.747982	0.758032
30000	0.762686	0.775165	0.795265

The effectiveness of DRDC performances was judged using True/False Positives (TP/FP) and False Positives/Negatives (FP/FN). Precision(PR) was calculated using PR= TP/ (TP+FP) while Recall (RC) was computed using RC = TP+FN. Table 4 lists the accuracy of the benchmarked methods.

Table 4 : Comparative accuracies of Benchmarked Methods

Method	Accuracy
NB	75.9378
SVM	78.0684
DRDC	80.32965

Table 4 clearly shows that DRDC performance is higher than other methods where it scores 80% in classifications compared to SVM's 78% and NB's 75%. Figure 11 depicts the precision time of DRDC.

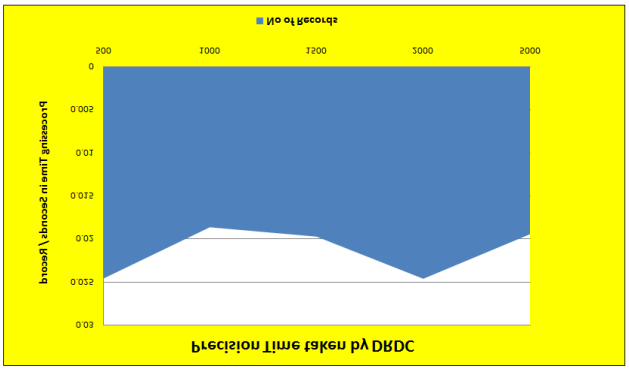

Fig. 11 : **Precision time of DRDC**

It is evident from the above figure that DRDC's processing time proportionally decreases as the record count increases. Figure 12 depicts the recall time of DRDC

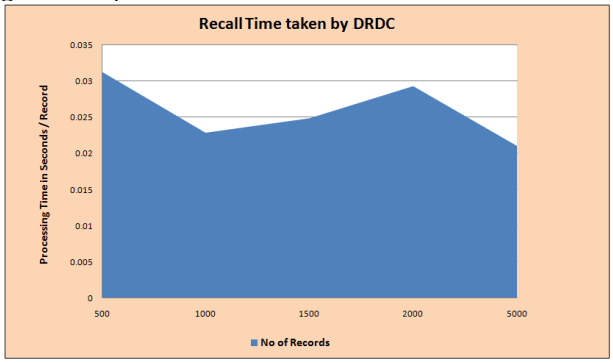

Fig. 12 : **Recall time of DRDC**

The above figure shows DRDC's performance increases with increasing record count. It takes comparatively the same Time or lesser for processing voluminous databases. Table 5 lists the Precision and Recall time values

Table 5 : DRDC Precision and Recall Times

No. of Tweets	Precision Time (in a sec)	Recall Time (in a sec)
500	12.33	15.66
1000	18.69	22.89
1500	29.66	37.33
2000	49.33	58.66
5000+	97.47	105.33

It can be seen from the above figure that the performance of DRDC does not decrease much as the number of tweets increases. It's precision and recall time increases proportionally based on the tweet count.

V. CONCLUSION

Behavioral Analysis is an interesting area for applying Natural Language Processing and automating conclusions from texts, typically for marketing trend analysis. Text mining can be used on unstructured text for extracting valuable information. This work has combined multiple DM techniques to identify user behavior from SM text, and its results are presented as figures and tables. Null values in words produce unimportant word sets and increase the processing time further. Hence they are discarded in this work. Training data on data with equal length is easy but training unequal length data is complex. Hence, this work uses DT first in its hybrid technique. The proposed technique has displayed its implemented results in the form of figures and tables that substantiate that DRDC is a viable technique and helps identify user behavior in BA. The results are easy to interpret while the technique is implementable. It can be concluded that DRDC is an automated technique that can be used on tweets BA.

VI. FUTURE ENHANCEMENT

Though this paper has proposed a combination of techniques for behavioral analysis, when applied to the sentimental analysis, the results might change. This research work would continue analyzing the sentiments of users, the basic expression of emotions and behavior. Behavioral Analysis of a market is an in-depth examination of business and customer spheres. It includes information pertaining to vital parameters of the industry. Future BA on customer behavior

can provide details about prevailing market trends, share, industry size, deliverables, and profit projections. Thus the future scope of this work is in extending BA to these parameters over some time.

REFERENCES

[1] https://www.brandwatch.com/sentiment-analysis-feature/.
[2] http://www.trackur.com/.
[3] https://netbasequid.com/blog/global-social-media-survey/
[4] E. Aydoğan, and M.A. Akcayol, ―A comprehensive survey for sentiment analysis tasks using machine learning techniques‖, In Proceedings of 2016 International Symposium on Innovations in Intelligent Systems and Applications (INISTA), Sinaia, Romania, (2016) 1-7.
[5] S. ChandraKala, and C. Sindhu, ―Opinion mining and sentiment classification: A survey‖, ICTACT journal on soft computing, 3(1)420-425, (2012).
[6] B. Agarwal, N. Mittal, P. Bansal, and S. Garg, ―Sentiment analysis using common-sense and context information‖, computational intelligence and neuroscience,2015, doi:10.1155/2015/715730.
[7] W. Medhat, A. Hassan, and H. Korashy, ―Sentiment analysis algorithms and applications: A survey‖, Ain Shams engineering journal, 5(4)(2014) 1093-1113 doi:10.1016/j.asej.2014.04.011.
[8] M. Tsytsarau, and T. Palpanas, ―Survey on mining subjective data on the web‖, Data Mining, and Knowledge Discovery, 24(3)(2012) 478-514 doi:10.1007/s10618-011-0238-6
[9] J. Bollen, H. Mao, and X. Zeng, ―Twitter mood predictsthestockmarket‖, Journal computational science, 2(2011)1-8, doi:10.1016/j.jocs.2010.12.007.
[10] T. Xu, Q. Peng, and Y. Cheng, ―Identifying the semantic orientation of terms using S-HAL for sentiment analysis‖, Knowledge-Based Systems, 35 (2012) 279289,doi:10.1016/j.knosys.2012.04.011.
[11] J. Brooke, M. Tofiloski, and M. Taboada, ‖Cross-linguistic sentiment analysis: From English to Spanish.‖, In Proceedings of International Conference RANLP-2009, Borovets, (2009) 50-54.
[12] Pang, B., and Lee, L. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2(2008) (1-21–135.
[13] Tumasjan, A.; Sprenger, T. O.; Sandner, P.; and Welpe, I. 2010. Predicting elections with twitter: What 140 characters reveal about political sentiment. In Proceedings of ICWSM.
[14] O'Connor, B.; Balasubramanyan, R.; Routledge, B.; and Smith, N. 2010. From tweets to polls: Linking text sentiment to public opinion time series. In Proceedings of ICWSM.
[15] Barbosa, L., and Feng, J. Robust sentiment detection on Twitter from biased and noisy data. In Proc. of Coling., (2010).
[16] Bifet, A., and Frank, E. 2010. Sentiment knowledge discovery in Twitter streaming data. In Proc. of 13th International Conference on Discovery Science.
[17] A. Hassan, A. Abbasi, and D. Zeng, Twitter sentiment analysis: A bootstrap ensemble framework, in Social Computing (SocialCom),2013 International Conference on. IEEE, (2013) 357–364.
[18] F. Coletta, N. F. F. d. Sommaggio Silva, E. R. Hruschka, and E. R.Hruschka Combining classification and clustering for tweet sentiment analysis, in Intelligent Systems, 2014 Brazilian Conference on. IEEE, (2014) 210–215.
[19] E. Kouloumpis, T. Wilson, and J. Moore, Twitter sentiment analysis: The good, the bad and the omg! ICWSM, 11(2011) 538–541.
[20] P. T. Ngoc and M. Yoo, The lexicon-based sentiment analysis for fan page ranking in Facebook, in Information Networking (ICOIN), (2014).
[21] A. Minanovic, H. Gabelica, and Z. Krstic, Big data and sentiment analysis using knime: Online reviews vs. social media, Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on. IEEE, (2014) 1464–1468.
[22] A. Porshnev, I. Redkin, and A. Shevchenko, Machine learning in predicting stock market indicators based on historical data and data from Twitter sentiment analysis, in Data Mining Workshops(ICDMW), 2013 IEEE 13th International Conference on. IEEE, (2013) 440–444.
[23] C. Troussas, M. Virvou, K. J. Espinosa, K. Llaguno, and J. Caro, Sentiment analysis of Facebook statuses using naive Bayes classifier for language learning," in Information, Intelligence, Systems and Applications (IISA), 2013 Fourth International Conference on. IEEE, (2013) 1–6.
[24] Saif M. Mohammad.#Emotional tweets. In Proceedings of the 1st Joint Conference on Lexical and Computational Semantics - Proceedings of the Main Conference and the Shared Task, and Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval'12). Association for Computational Linguistics,1(2)(2012) 246–255
[25] Asmi, A., Ishaya, T., Negation identification and calculation in sentiment analysis. In The Second International Conference on Advances in Information Mining and Management, 1-7. (2012)
[26] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in Soviet physics doklady, 1966, p. 707.].
[27] S. G. J. Vargas, A Knowledge-Based information Extraction Prototype for Data-Rich Documents in the Information Technology Domain, National University, (2008).
[28] G. Navarro, A guided tour to approximate string matching, ACM Computing Surveys(CSUR), 33, (2001) 31-88.
[29] J. F. Daoason, Post-Correction of Icelandic OCR Text,(Master's thesis, University of Iceland, Reykjavik, Iceland), (2012).
[30] I. Q. Habeeb, S. A. Yusof, and F. B. Ahmad, Two Bigrams Based Language Model for Auto-Correction of Arabic OCR Errors, International Journal of Digital Content Technology and its Applications,8(28) (2014) 72- 80.
[31] I. Q. Habeeb and S. A. Yusof, Design of Automatic Bilingual Lexicon for Arabic OCR Post-Processing Errors Correction, in International Conference on Rural ICT Development, Malacca, MALAYSIA. (2013).
[32] C. J. van Rijsbergen, (1989), Information Retrieval, Buttersworth, London, second edition, Gerald Kowalski, Information Retrieval Systems – Theory and Implementation, Kluwer Academic Publishers, (1997).
[33] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey, Scatter/Gather: ACluster-based Approach to Browsing Large Document Collections, SIGIR 92(1992) 318 – 329.
[34] Oren Zamir, Oren Etzioni, Omid Madani, Richard M. Karp, Fast and Intuitive Clustering of WebDocuments, KDD.,97(1997)287-290.
[35] Daphe Koller and Mehran Sahami, Hierarchically classifying documents using very few words, Proceedings of the 14th International Conference on Machine Learning (ML), Nashville, Tennessee, (1997) 170-178.
[36] Basavesha D, Dr. Y S Nijagunarya. Soft Computing based Duplicate Text Identification in Online Community Websites, International Journal of Engineering Trends and Technology 68(7)(2020) 1-7.