Text filtering on Social Network Based Text Classification

Dr M.Surya Bhupal Rao1, Dr S. Rahamay Basha2,
*Dept. of Computer Science and Engineering*
, CVR College of Engineering 1,Mangalpalli, Ibrahimpatnam,Hyderabad

MRCET2, Hyderbad

 Hyderabad, India

suryabhupal@gmail.com
basha.ste@gmail.com

***Abstract—*In day to day communication between individuals and organizations, Social Networks(SN) are growing important. The SN mining has captured the attention of researchers for various purposes such as marketing, identification of hot topics and events. But effectively mining SN data is a difficult task as it includes a lot of noisy information (e.g. meaningless information, slang term, etc.,). It is important to filter out noisy information for successful social network data mining. The proposed collection of features should reduce the negative-impact on the classification of short texts of the noisy and sparse features of social network data. Model is trained-with traditional Machine-Learning(ML) methods in the-field of Artificial Neural Networks (ANN), K-Nearest Neighbor (KNN), Naïve Bayes (NB), Decision Trees (DT), Support for the Vector Machine (SVM). Experiments on the social network data set show that there are obviously advantages compared to standard alternatives to the approach proposed which may improve the semantic association between classification results and category tasks. Hence, the proposed method will achieve root mean square error(RMSE), precision, recall, F-score and accuracy is analyzed and evaluated.**

Keywords—Short Text, Text Classification, Hyper Parameter Tuning, Metrix, Social Network.

# Introduction

Text classification is the method by which the text document with most relevant labeling and categories is automatically tagged [1]. Most texts to be classified in the Social Network are ambiguous, i.e. one instance that belong to multiple labels. Social networks(SN) such as Twitter, Facebook and YouTube made access to news, activities, and other information much easier for people. Massive content on the SN is commonly considered as very important as it includes various useful information including real-time news, hot activities, popular topics, etc. Hence, a majority of industry and academics have attracted realistic research to derive relevant knowledge from massive SN data.

The short term is one of-the-most popular forms of SN data, such as 'tweet' or 'post'. We do use the term "post" to make the description more convenient. The character-limit for each message is constrained by the services, for example, Twitter does not accept lengthy messages with longer than 140 characters. The SN classification attempts to accumulate useful data and remove noisy data from the bulk of data or identify short texts in a collection of previously identified topics. This is a fundamental technique for a wide-variety of activities related to social networking such as hot topic tracking [7] in real-time event detection [4, 5, 6] and analysis of user relations [8].

The short text classification of social network is a text classification type. But compared to text classification, it is more challenging. In the social network the limited number of words and various other non-standard words make the corresponding short texts unclear and ambiguous. Hence the main problems to be solved in the short text classification area are how to successfully use the restricted textual information and overcome the impact of noisy information.

The researchers worked, through ensemble learning, to improve short text representation by integrating several types of features. However, they did not discuss the problem of providing specific capacity for learning with various kinds with features based on machine learning, which may impair the-performance-of the proposed techniques. In this work proposes new SN classification technique based on a multi-attribute feature extraction method.

 The remaining of this article is organized as. The related work is highlighted in Section II. The methodology is clarified in Section III. The experimental results and evaluation are analyzed in Section IV. The final research is concluded by Section V.

# Related Work

Social networking platforms such as Twitter, -Facebook, -LinkedIn, and YouTube-are-the internet's-most-popular-sites for all age-group. This platform provides a social link to many users. The users of these websites share, organize teams and provide usable information. As people post information on websites through social-network, they generally share their views and emotions. In this way, the data-collected from the online-conversation-may be more credible and unfiltered than responses to formal inquiries. Such contextual menus act as a background for the experiences of users [13]. The whole information includes a powerful meaning that classifies person's regular activities as daily updates, views, thoughts, opinions, image-based feelings, smiley and experiences.

The SN is the framework for the social-relationships that have common preferences and the location of information and knowledge shared by users. Filtering information in SN is used to prevent the posting or viewing of inappropriate comments on user walls. For classification, different forms of machine learning are used. Extracting information from views exchanged in web forums, blogs, opinion polls and comment boxes is a technique of opinion mining [14]. However, opinion mining uses techniques of natural-language-processing and text-mining to allow computers to understand emotional speech. Moreover, its principal concern is the retrieval, from the unstructured text, of sentimental and emotional expressions [15]. The discovery of the best classification method is a crucial task to evaluate feelings. For opinion analysis, multiple methods use the corpus [16, 17]. Social networking content provides wonderful opportunities to express happiness and pain, access emotions and stress, and seek social support. Users chat and post their day-to-day encounters informally and unintentionally in various social media platforms. Through the advancement of social media platforms, people can share their thoughts and opinions. Our main purpose is to analyze and take into consideration their-advantages-and-disadvantages the different classifiers used to identify messages.

The research-has-been implemented to analyze the role of the TF-IDF algorithm development method to gather data directly from Twitter servers. The implementation consists of a Twitter 2.0 web interface. As the aim of the analysis was to take tweets from Samara region, the Samara region (within the region's settlements) was chosen as a criterion for the geolocation of messages. For short messages of 140 characters’ length and the k-mean algorithm, the TF-IDF metric includes cover tweets received in this way.

Ensemble learning is a means to solve common problems of machine learning. To achieve predictive accuracy, the ensemble methods use many machine learning algorithms. Unlike ML methods, ensemble approaches do not apply to any particular paradigm of learning. Ensemble-learning is used to allocate confidence to a model's decision, to select appropriate features, to train incrementally for best results and to correct errors. Combining different classification models to increase the outcomes of the classification is developed by use of the ensemble-based method. Common learning process algorithms are bagging (bootstrap-aggregate), boosting, stacking. Boosting and bagging is a training method for week models. These approaches by aggregating-the-predictions of many further sub-classifiers produce class predictions.

Other conventional ML models are used as ANN, NB, KNN, DT, SVM, Logistic-Regression, Fuzzy Logic, Genetic Algorithm [9, 10, 11, 12], to train the data set. The team focused on various data mining methods to identify efficient approaches to issues with the Social Network. Boosting approaches are used by authors in this analysis as XGboost and Catboost algorithms. An optimal selection of parameters is required to improve the performance of such algorithms. For the same we use selection method of hyper parameters. Other conventional ML approaches are also used to recognize short texts and compare results on social media.

# Methodology

Here we describe the entire process of work from data collection to performance as seen in Figure 1. The goal of social network post-classification is to obtain valuable information from the massive data generated or categorized by social networks into a set of specific topics by short post texts.  figure 1 Shows the overall method of short text classification. After   preprocessing of data and extraction of features, the text samples are transformed into fixed-length vectors, which are used to train a classification model. The model training phase adopts several supervised ML algorithms to produce a final short text classifier. Hence the trained classifier will be used to determine the label of the new post when it comes to a new unlabeled comment.



1. Training and testing data with ML models is further developed to improve performance with hyper parameters and to compare results of all ML models.

## ANN Learning

The network contains a broad variety of deeply interdependent computing components (neurons), which work together to solve every specific problem. When they can extract useful information from a large number of dataset, neurons for particular applications like, feature extraction, noise reduction and pattern recognition have been configured. The relation between two neurons throughout the neural network determines the strength of one neuron on the other. Although the connecting weight determines authority strength between the two neurons. In neural networks there seem to be two methods such as supervised and unsupervised learning. The neural network is trained in supervised learning using a collection of inputs and required performance patterns given by an investigator [18]. Researchers are becoming increasingly popular in texts mining due to the huge amount of text accessible via social media, such as forums, blog posts, chat rooms, digital libraries and communities. For logical text processing on social Websites the neural network may be used.

## Naïve Bayesian Classifier

In text classification frequently used classifier is NB [12]. The basic concept behind this classification is to find a probability of which class belongs to this document. Through using these feedbacks collected from multiple social media sites we can understand the profiles. It is basic, but also results in more-sophisticated-methods-of classification. Maximum probability estimates the model parameters. To measure parameters, it takes a small amount of training. It works in supervised learning well and effectively. The order of rank of the pages is classified here. The text classification is based on the post-probability estimate for the documents contained in various classes. NB is based on the theorem of Bayesian independence feature selection. For anti-spam filtering, the NB system is used. It is-divided-into two stages. The first stage has been available for the data set and the second phase is classified.

Prior likelihood of NB analysis: this is a hypothesis based on experience achieved previously. This is a measure of the number of individual objects and total objects. Likelihood: In which case it is categorized as a new object. Posterior Probability: The final categorization occurs by integrating all information sources.

##  KNN classifier

K-nearest-neighbor classifier is based on principles that are similar points (documents) belonging to the same class of space. It determines the similarity with each neighbor between the test document. KNN is a case-based learning algorithm built on the distance or similarity of a several observations. such as the-Euclidean-distance or similarity measurements for Cosine [3,12]. Due to the efficiency, nonparametric and easily implementable properties, many applications use this method. The classification time is however long and difficult to reach the optimum value of k. The best choice of k is based on the information. Specific heuristic techniques will select a good k. We are using the same major issue and dataset used in logistic regression to apply the Python K-NN algorithm. But we are now improving the model's performance.

## Support Vector Machine(SVM)

 The SVM help considers the linear splitting hyperplane that maximizes the margin, i.e. Nonlinear case separable:  kernel function and Hilbert space. Positive and negative teaching is required for the SVM as it is uncommon for-other classification-methods [18]. For SVM to analyze the decision area which best differs between positive and negative data-in the n-dimensional space, called the hyperplane, these-positive and negative-training sets are required. d A support vector is represented the document member who is nearest to the decision surface.

## Decision Tree(DT)

The DT consists-of a root-node comprising all documentation for text classification. Each node is a subset of documents divided by an attribute. Every arc has a labeled that can be used for the parent attribute. A class is labelled on every leaf node. The hierarchical decomposition of the data space was designed. The predicate or condition is determined-according to the reference value. Pruning    is to be done, in order to reduce over fitting information. The listed-splits are possible in-the decision-trees for many different forms of splits. Single split feature, Multi-attribute classification based on similarities, Dimensional multi-attribute split. They are generally applied in a text context, as compared with ID3, C4.5 for text classification [2] in minor variations.

***1)*** *Bagging*

The composition of the inputs includes bagging or bootstrap aggregation. The bagging classifier is adaptable to any of the simple classifiers in the abnormal subsets of the first dataset and then their individual preferences can be summed by casting a vote or averaging the final prediction. The Random Forest is a stronger technique than bagging Only selected group of Features are selected randomly in random forest.

***2)*** *Boosting*

Boosting makes reference to a community of algorithms that translates poor learner to high learners. Boosting is a community approach that improves the predictions of a provide learning algorithm. The idea is to sequentially train weak learners, each of whom tries to correct their predecessor. Boosting algorithms offers ML models superpowers to improve the precision of their predictions. Gradient Boosting Machine (GBM) integrates several decision trees' predictions to generate actual predictions.

***3)*** *XGBoost algorithm*

XGBoost stand Extreme gradient boosting. XGBoost is a machine learning algorithm focused on the decision-tree which uses a boosting gradient framework. XGboost relies on the performance-and-speed of the training models [19],[20]. We also use the Catboost algorithm to know training performance for other boosting algorithms and achieve accuracy. CatBoost algorithm is further modified to have maximum accuracy for the best learning parameters.



1. XGBoost algorithm Evolution

## HYPERPARAMETER TUNING (HPT)

HPT is the selection process which determines the optimized architecture of models for accuracy. A hyper parameter is a parameter that controls the learning process using its value. hyper-parameter tuning approaches are Grid-search and Random-search. The most frequently used grid search, applied to data. Hyperparameter optimization is performed by grid search, it is common way. Grid search is worked through a specified hyperparameter subsets by searching exhaustively. The advantage in grid search is that the optimal combination supplied parameters are assurance found. We use the embedded 'GridSearchCV' search method from the science-learning python Library to identify suitable hyper-parameters for improved performance. The advantage of grid search being that the optimal combination parameters given is guaranteed to be identified. The disadvantage is time consuming and expensive.

The random search is different from the grid search. The main thing is, it searches the hyperparameter specified subset randomly. The major advantage is reduced the processing time. There is an improve to reduce the processing time, but We are unable to find the hyperparameter optimal combination. Here we discussed two methods of turning hyperparameter and we saw model performance improvements. While this is an essential step in modeling, the best way to boost performance by no way. We may discuss certain forms in future articles to avoid overfitting, like the and ensembling and feature selection.

## Metrics

### Root Mean Square Error(RMSE)

The rmse is a common measure of variations between the values expected by a model, and the values actually observed. When developed a successful pattern, the RMSE for the training and test sets would be extremely similar. When the RMSE of the test collection is far higher than the RMSE of the training, then may have skipped the data

$RMSE=\sqrt{(E-O)^{2}}$(1)

Where-E=>expected, O=>observed value. As seen in equation 1, RMSE can be calculated.

### Precision (Positive-Predicted Value)

The number-of false-positives would be limited by precision. It checks whether the positive result is predicted by a classifier. It is determined that the number-of true-positive predictions-divided by the-number of cumulative positive predictions as seen in equation 2 can be estimated. The ability to return only specific instances is that with a classified model.

$Precision=\frac{TP}{TP+FP}$ (2)

where TP=>True-Positives, FN=> False-Negatives.

### Recall (True-Positive Rate)

It calculates sensitivity and is determined as the true positives number predictions-divided by the total-number of positives predictions, according to equation 3. Best sensitivity is 1 and lowest sensitivity is 0. The ability to classify all relevant instances is a classification model.

$Recall=\frac{TP}{TP+FN}$ (3)

where TP=>True-Positives, FN=> False-Negatives.

### F-score

F-score offers a means of combining precision and recall into one metric that combines all properties. The two scores will be combined into computation of the F score after precision and recall have been determined for a multiclass or binary problem of classification. Single metric uses the harmonic-mean in equation 4, which combines recall and accuracy.

$F-score =( 2\*Precision \* Recall) / (Precision + Recall)$ (4)

### Accuracy

The accuracy of the Machine Learning Algorithm is the approach used to decide which algorithm determines the best relationships and patterns between variables in an input or training data dataset. When a model is more common of 'unseen' data, clearer forecasts and predictions may be produced that provide greater business value. As seen in Equation 5, accuracy can be determined.

$Accuracy=\frac{TP+TN}{TP+FP+TN+FN}$ (5)

True positive and true negatives are the correct predictions of the model. Many such predictions have been True-Positive (TP), True-Negative (TN), False-Positives (FP) and False-Negative (FP), respectively.

# Experimental Evalution

## Datasets

Two real-world tweet data sets called dataset-1 and data-2, respectively, are used in our experimentation. The Dataset-1 was for the binary classification method, which was structured to determine whether or not a tweet was related to an event (this was the same feature for Twitter Stand [28]). We provided label 1 to a tweet if it was linked to an event that suggested that the tweet was a useful text. This dataset included 5000 manually labeled messages, crawling between June 2017 and May 2017 utilizing Twitter Streaming API.

In addition, the label 0 was applied to the tweet and the tweet was shown to be useless. We have labelled 2500 valuable tweets and 2500 useless tweets with polarity labeling to indicate that the classification model was not disturbed by a data balancing problem. Dataset-2 for the classification task was multiple classes. We established four topic-related class labels for this dataset: education, economics, military and sports. The Twitter REST API from 500 machines manually labeled about 6000 tweets, with around 1500 tweets per class.

## Preprocessing

For several OSN text mining activities, pre-processing is a required step [21, 22, 23]. We have used the Python National Language Toolkit (NLTK) [24] to preprocessed our data in this document. Our experimental preprocessing consisted of three phases: tokenization, part-of-speech (POS) and Named Entity Recognition(NER)[25]. Each tweet was performed to a bag of words for tokenization. Separate strings such as URLs, username and # hashtags have been saved. As input of the other measures we stored words, punctuations and numbers.

Afterwards each word was described in the form of $tok, tags$, using a pretrained POS tagger, where tok is a word or symbol, and tag is the related part-of-speech tagging result. In addition, each word is archived by a pre-trained NER model with three separate entities: the LOC (location), the PER (person name) and the ORG (organisation). As a result, each word can represented by a 3-tuple $(tok, tag, ner$), where $ner$ is the relevant named-entity of the $tok$. When the named-entities were uncommon, $(tok, tag, ner)$ was reduced to $(tok, tag)$ to save the memory space. When ner takes any PER, ORG and LOC values, then a tag attribute is transferred to named entity from part of speech. Otherwise, word as part of speech, we kept the tags value. The preprocessing details can be found in the Figure 1. That tweet was defined by a tuple $[(toks\_{1},tag\_{1}), (toks\_{2},tag\_{2}), \\_ \\_ \\_ , (toks\_{M}, tag\_{M})]$ sequence after preprocessing.

## Results and Analysis

In these research, shown in Table 1, the data set is classified into training set, validation set and test set, based on a ratio of 8:1:1. The total number of samples in the data collection is shown in the total number of samples. The average SN word per sample is Words/Sample. Label/Sample refers to the average multi-label categorized number of labels per sample, and Levels/Sample refers to an average hierarchical sampling classification.

 TABLE I SUMMARY OF DATASETS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data set | Total | Words/ | Labels/ | Levels/ |  |
| Samples | Sample | Sample | Sample |  |
|  |  |
| training set | 101,914 | 182 | 2.3 | 6.1 |  |
| validation set | 12,814 | 202 | 2.3 | 5.6 |  |
| test set | 12,814 | 197 | 2.3 | 6.3 |  |

These are the hyperparameters used in the experiments. The term "embed" is 128, the sentence length is 600, the LSTM is bidirectional and the size of the batch is 32, the optimizer uses the algorithm Adam, the rate of learning is 0.001, the betas is 0.9 and the eps are 1e-08, weight decay is 0.5 and 0.5 for teacher forcing ratio. The prediction is based on the search technique Beam Width set to 3. In this study, alternative multi-tasks training is used to execute multi-tasks by naming each task's optimizer in alternative.

The experimental effects of model performance and baseline approaches to the dataset show in Tables II-III for multi-label classification. Therefore, the performance of classification is shown by adding the hierarchical structural mask matrix.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | TABLE II TEXT MULTI-LABEL CLASSIFICATION RESULTS |  |  |  |  |  |  |  |
| Model |  | Training Set |  |  |  |  | Validation Set |  |  |  | Test Set |  |  |  |
|  | *Accuracy* | *PrecisionF* | *Micro-F1* | *Accuracy* | *PrecisionF* |  | *Micro-F1* | *Accuracy* | *PrecisionF* | *Micro-F1* |
|  |  |  |
| ANN |  | 0.902 | 0.873 |  | 0.869 |  | 0.912 | 0.873 |  |  | 0.869 |  | 0.902 | 0.873 |  | 0.869 |  |
| KNN |  | 0.944 | 0.941 |  | 0.924 |  | 0.964 | 0.941 |  |  | 0.924 |  | 0.954 | 0.941 |  | 0.924 |  |
| Naïve Bayes |  | 0.964 | 0.943 |  | 0.907 |  | 0.944 | 0.933 |  |  | 0.917 |  | 0.954 | 0.933 |  | 0.917 |  |
|  SVM | 0.962 | 0.946 |  | 0.926 |  | 0.962 | 0.936 |  |  | 0.916 |  | 0.952 | 0.936 |  | 0.916 |  |
|  Decision Tree |  | 0.969 | 0.950 |  | 0.929 |  | 0.949 | 0.940 |  |  | 0.923 |  | 0.959 | 0.940 |  | 0.923 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | Table III TEXT HIERARCHICAL CLASSIFICATION RESULT |  |  |  |  |  |  |  |
| Model |  | Training Set |  |  |  |  | Validation Set |  |  |  | Test Set |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | *Accuracy* | *PrecisionF* | *Micro-F1* | *Accuracy* | *PrecisionF* |  | *Micro-F1* | *Accuracy* | *PrecisionF* | *Micro-F1* |
|  |  |  |
| ANN |  | 0.921 | 0.764 |  | 0.886 |  | 0.901 | 0.736 |  |  | 0.871 |  | 0.781 | 0.644 |  | 0.799 |  |
|  KNN | 0.911 | 0.785 |  | 0.881 |  | 0.893 | 0.771 |  |  | 0.877 |  | 0.813 | 0.685 |  | 0.846 |  |
| Naïve Bayes |  | 0.922 | 0.798 |  | 0.913 |  | 0.901 | 0.799 |  |  | 0.879 |  | 0.834 | 0.722 |  | 0.862 |  |
|  SVM | 0.931 | 0.804 |  | 0.925 |  | 0.904 | 0.799 |  |  | 0.871 |  | 0.839 | 0.728 |  | 0.835 |  |
|  |  |  |  |  |  |  |
|  Decision Tree |  | 0.959 | 0.846 |  | 0.928 |  | 0.928 | 0.805 |  |  | 0.898 |  | 0.845 | 0.752 |  | 0.842 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Comparing with the State-of-the-art Works:**

To demonstrate the benefits of multi-attribute features extracted from this study. Three state-of-the-art models for short text representation were compared to the proposed method: BOW [14], word2vec [24] and 8F model [25]. We conducted experiments using four approaches on both dataset-1 and dataset-2. We select classifiers which are able to describe the features in the BOW, word2vec and 8F models. Similarly, to determine the performance of four approaches, accuracy, recall, precision and F1 score are used.

From fig.10 represents the compared performance of four approaches using the data set-1 in a binary classification task. We can see from the figure that the BOW is worst performance since the short text is just classified using the words. As we know, OSN short text normally includes very few words, so the acceptable word bag does not well reflect the short text.

In terms of accuracy 8% higher than the BOW model, the word2vec and 8F model achieve approximately the same efficiency. The reason, that both approaches use more information in short texts (e.g. semantic and structured information). Our system was around 9% above state-of-art work in terms of all four metrics relative to the word2vec and 8F versions.

Figure 3. Performance comparison between different methods under binary class classification task.

Figure 4 . Performance comparison between different methods under multiple class classification task.

We have measured the accuracy of the four OSN short text classification approaches with the use of dataset-2 for multi-class classification. We have compared four approaches in the Fig 11. The performance comparison we observed from the figure that the 8F model was the worst possible. Features extracts simply 8F model from short text statistics and doesn’t contain any topic associated information such that the 8F model failed to produce an acceptable classification result in the multiclass classification task. BOW and word2vec are performed better than 8F and the word2vec performed better than BOW. Word2vec extracts semantic word information in context, while the BOW only takes into consideration words information. The method proposed performed is better of all methods. The feature set proposed includes features of semantic attributes that can represent topics of text. The social feature attribute and structure attribute of the short text can be influenced by the relevant research topic in OSNs, thus these two categories of features often include details that may help to enhance the efficiency of short text classification.

The results demonstrate that the proposed method as an ensemble was clearly better than the simple classificators in OSN short text classification. We subsequently tested the efficacy of the feature set by comparing the BOW, word2vec and 8F variants of the three modern short text representation models. The experimental outcomes from binary and multi-class classification tasks have revealed that the proposed feature set would significantly improve the precision of OSN's short text ratings. Therefore our approach can be used in several real-world short text classification tasks and can produce adequate classification results through integrating the ensemble technique with the multi-attribute features.

Performance Evaluation Metrics: We use many metrics, including RMSE, Precision, Recall, F- score and Accuracy for evaluating various classification-models to determine-the-performance of the method proposed in detail. Test results was collected from the label tweets in our experiments. The tweet is described as the positive sample and the tweet is identified as the negative sample. For tweet in the test data, we-claim that the tweet is classified correctly, because the label predicted by the classifier is the same as the manually assigned label. The accuracy is described by the ratio of the correctly estimated number of tweets to the total-number of test tweets. The recall is used to assess the right estimates about as many positive samples are positive samples. Precision is used to calculate the percentage of valid tests that are truly accurate.

Figure 5 RMSE for Machine Learning approaches are applied.

Figure 6 Precision for machine learning approaches are applied.

Figure 7 Recall values for machine learning approaches are applied.

Figure 8 F-Score for machine learning approaches are applied.

Figure 9 Accuracy for machine learning approaches are applied.

First, a novel feature set to accurately define the OSN short texts was developed to conquer the noisy and sparse features of OSN short texts, taking the specific social attributes, structure attributes and semantine attribute information completely into account. Secondly, an ensemble classifier of multiple regression models has been developed to completely leverage knowledge in the derived feature collection. For realistic OSN short text activities, the proposed methodology has obtained satisfactory results. After applying the proposed method to a challenge to identify actual events we found that our system could greatly increase the efficiency and the quality of useful short text filtering by contrasting it with specialised approaches.

Figure 10 ML models according to different Performance measure

# Conclusion

This paper discusses the problem of how valuable data can be filtered from the vast SN text which is important to many SN data mining tasks including in real-time event monitoring, hot topic monitoring and evaluating user relationships. After validating and preprocessing the dataset then we applied ANN KNN, NB, DT, XG-Boost and Cat-Boost methods with HPT and without HPT. The results are accepted that the data is efficiently trained and tested using the HTP CatBoost method. Within this study, we analyze the results for the different performance parameters of the social network among these learning methods. In our proposed approach to achieved better results for short text on Social media. After implementing the method proposed for real world event identification, we find that by contrasting the state-of-the-art methods our solution would achieve substantially improved efficiency by obviously improving the quality of useful short text filtering.

##### References

1. Ashraf Elnagar , Ridhwan Al-Debsi, Omar Einea. Arabic text classification using deep learning models. Information Processing and Management, 57(1), pp.1-17(2020).
2. K. Nirmala, S. Satheesh kumar and Dr. J. Vellingiri “A Survey on Text categorization in Online Social Networks” International Journal of Emerging Technology and Advanced Engineering Volume 3, Issue 9, September 2013.
3. Vandana Korde, C Namrata Mahender “TEXT CLASSIFICATION AND CLASSIFIERS: A SURVEY” International Journal of Artificial Intelligence & Applications (IJAIA), Vol.3, No.2, March 2012.
4. C. Comito, Carmela, A. Forestiero and C. Pizzuti, “Bursty Event Detection in Twitter Streams,” ACM Transactions on Knowledge Discovery from Data, vol. 13, no. 4, pp. 1–28, Aug. 2019.
5. M. Fedoryszak, B. Frederick, V. Rajaram and C. Zhong, “Real-time event detection on social data streams,” in Proc. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining(KDD’19), Anchorage, AK, USA, 2019, pp. 2774–2782.
6. J. Nichols, J. Mahmud and C. Drews, “Summarizing sporting events using twitter,” in Proc. Proceedings of the 2012 ACM international conference on Intelligent User Interfaces(IUI’12), Lisbon, Portugal, Feb. 2012, pp. 189–198.
7. S. Rahamat Basha, J. K. Rani, “A Comparative Approach of Dimensionality Reduction Techniques in Text Classification”, Engineering, Technology & Applied Science Research, Vol.: 9, Issue: 6, pp. 4974-4979, 2019.
8. C. Tan, L. Lee and J. Tang, “User-level sentiment analysis incorporating social networks,” in Proc. Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining(KDD’12), San Diego, California, USA, Aug. 2011, pp. 1397–1405.
9. X. Liu, Z. Liu, G. Wang, Z. Cai and H. Zhang, "Ensemble Transfer Learning Algorithm," in IEEE Access, vol. 6, pp. 2389-2396, 2018.
10. Yousry Ibrahim, Salah Kamel, Ahmed Rashad, Loai Nasrat, and Francisco Jurado, “Performance Enhancement of Wind Farms Using Tuned SSSC Based on Artificial Neural Network,” International Journal of Interactive Multimedia and Artificial Intelligence, vol. 5, no. 7, pp. 118–124, May 2019.
11. Qinghan Xue Mooi, and Choo Chuah, “Explainable deep learning based medical diagnostic system,” Smart Health, vol. 13, pp. 1–17, 2018.
12. S. R Basha, et al , “Impact of featureselection techniques in Text Classification: An Experimental study”, J. Mech. Cont. Math. Sci., Special Issue, Issue: 3, pp. 39-51, 2019.
13. Xin Chen, Mihaela Vorvoreanu, and Krishna Madhavan “Mining Social Media Data for Understanding Students’ Learning Experiences,” IEEE transactions on learning technologies, manuscript id 1, (2013), pp. 1-14.
14. S. Rahamat Basha, J. K. Rani, J. J. C. P. Yadav, “A Novel Summarization-based Approach for Feature Reduction Enhancing Text Classification Accuracy”, Engineering, Technology & Applied Science Research, Vol.: 9, Issue: 6, pp. 5001-5005, 2019.
15. Khan, K., B. Baharudin, and A.Khan. Mining opinion from text documents: A survey. In Digital Ecosystems and Technologies, 3rd IEEE International Conference on. 2009: IEEE: pp. 217–222.
16. Surya Bhupal Rao, S. Rahamat Basha, G.Ravi Kumar, “A Comparative approach of Text Mining: Classification, Clustering and Extraction Techniques”, Journal of Mechanics of Continua and Mathematical Sciences, Special Issue, No.-5, January (2020) pp 120-131, DOI:[https://doi.org/10.26782/jmcms.spl.5/2020.01.00010](http://www.journalimcms.org/special_issue/a-comparative-approach-oftext-mining-classification-clustering-andextraction-techniques/)
17. Valitutti, C.S.a.A., WordNet-Affect: an affective extension of WordNet. In Proceedings of 4th International Conference on Language Resources and Evaluation, 2004: pp. 1083–1086.
18. Rizwana Irfan, Christine K. King, Daniel Grages, Sam Ewen, Samee U. Khan, Sajjada. Madani, Joanna Kolodziej, Lizhe Wang, Dan Chen, Amma R Rayes, Nikolaos Tziritas, Cheng - Zhong Xu, Albert Y. Zomaya, Ahmed Saeed Alzahrani, And Hongxiang Li “A Survey on Text Mining in Social Networks, ” The Knowledge Engineering Review, United Kingdom, (2004) pp.1-24.
19. Vikrant A.DevMario R.Eden, "Formation lithology classification using scalable gradient boosted decision trees," Computers & Chemical Engineering, vol. 128, pp. 392-404, September 2019.
20. Sivala Vishnu Murty, R Kiran Kumar, "Accurate Liver Disease Prediction with Extreme Gradient Boosting ," International Journal of Engineering and Advanced Technology (IJEAT), vol. 8, no.6, pp. 2249–8958, August 2019.
21. S. Wankhede, R. Patil and S. Sonawane, “Data Preprocessing for Efficient Sentimental Analysis,” in Proc. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), April. 2018, pp. 723–726.
22. S. B. Kotsiantis, D. Kanellopoulos and P. E. Pintelas, “Data preprocessing for supervised leaning,” International Journal of Computer Science, vol. 1, no. 2, pp. 111–117, 2006.
23. M. Van de Kauter, G. Coorman and E. Lefever, “LeTs Preprocess: The multilingual LT3 linguistic preprocessing toolkit,” Computational Linguistics in the Netherlands journal, vol. 3, pp. 103–120, 2013.
24. G. Ravi Kumar, S.Rahamat Basha, Surya Bhupal Rao, “A Summarization on Text Mining Techniques for Information Extracting from Applications and Issues”, Journal of Mechanics of Continua and Mathematical Sciences, Special Issue, No.-5, January (2020) pp 324-332. DOI:[https://doi.org/10.26782/jmcms.spl.5/2020.01.00026](http://www.journalimcms.org/special_issue/a-summarization-on-text-mining-techniques-for-information-extracting-from-applications-and-issues/).
25. A. Ritter, S. Clark and O. Etzioni, “Named entity recognition in tweets: an experimental study,” in Proc. Proceedings of the conference on mpirical methods in natural language processing(EMNLP), Edinburgh, UK, July. 2011, pp. 1524–1534. M. Luckner, B. Topolski, and M. Mazurek, "Application of XGBoost algorithm in fingerprinting localisation task," in Proc. IFIP Int. Conf.Comput. Inf. Syst. Ind. Manage. Cham, Switzerland: Springer, pp. 661–671, 2017.