INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 7, Issue 2 - V7I2-1523)
Available online at: https://www.ijariit.com

A comparative analysis of machine learning techniques for automatic text classification

Sivakami M.
sivakamimk@gmail.com
Madurai Kamaraj University, Madurai, Tamil Nadu

Dr. M. Thangaraj
thangarajmku@yahoo.com
Madurai Kamaraj University, Madurai, Tamil Nadu

P. Aruna Saraswathy
arunasaraswathyp@gmail.com
Madurai Kamaraj University, Madurai, Tamil Nadu

ABSTRACT

Text processing and its related activities have reached its peak demand in the present days due to the increase of unstructured data. The underlying structure in any text can be derived through categorization techniques. The capacity of text classification algorithms to perform the conversion from structured to unstructured data is the key factor in all text processing activities. To further enhance this, many concepts from other disciplines such as statistics, physics and mathematics were tailored to suit the needs of text analyzing pipelines. Text classification techniques help to build the template necessary for extracting meaningful information. Hence, this paper undertakes a study of comparison on various text classification algorithms to reiterate their suitability for particular classes of problems. The algorithms such as 'Naïve Bayes', 'Support Vector Machine', 'K- nearest neighbour' and 'Decision Tree' were studied based on empirical analysis with respect to WEKA data analysis platform. From the experimental results, it is seen that the strength of algorithms depended on the data type, nature of attributes and representation of the classes. This is verified by various accuracy metrics used in the study such as precision, recall, accuracy, F1- scores and ROC values.

Keywords: *Naïve Bayes, Support Vector Machine, Decision Tree, Text Classification, Weka, J48, Automatic Text Mining, IBK*

1. INTRODUCTION

The studies related to text mining are on the rise due to availability of number of documents that come from variety of sources. They are majorly two types structured and unstructured data [9]. The interesting finding is that, ninety percent of data generated is in unstructured format.

Text mining is the major field concerned with text classification, it deals with classifying new user or entity or document in to predefined categories [2]. The categories are predicted based on supervised or unsupervised techniques. The categories can be binary such as 'yes' or 'no', multiclass, where entities are classified in to more than two classes or multi-labels where a single entity can be classified into more than one class. Integrating concepts and techniques from other similar domains such as Information Retrieval, Statistics, Machine learning have also shown to produce promising results in automated Text Classification (TC).

The goal of TC is to assign a given document or element to the appropriate category by analyzing the words in the document [4]. It has numerous applications in the form of scientific articles, document genre identification, spam filtering, authorship categorization, survey grading, etc.

The field of Machine Learning relates to designing and development of techniques that are able to learn the features and use the knowledge to predict the categories for unforeseen instances. This paper analyzes various such algorithms as, Naïve Bayes classifiers (NB), Support vector machine (SVM) and k-Nearest Neighbors (kNN) for their ability in classification frameworks using prominent datasets.

The rest of the paper is organized as section 2 deals with survey of related work was carried out to identify the strengths and weakness of the algorithms, section 3 deals with explanation about the algorithms, section 4 discusses the empirical results and section 5 concludes with solutions and future research.

2. RELATED WORK

The work on text classification is majorly undertaken in social networks and document classification tasks. The meta-information and other information about the entities are helpful to derive features and use them for further classification in many existing works. There are multiple ways to carry out text categorization tasks, as specified by [1]. In [16] Support Vector Machine is used for solving text classification problem. Interestingly, the paper uses, voting to optimize results. Bayesian interface can be used for text classification [14]. The Information Gain (IG) and Maximum Discrimination (MD) were used for improving feature representation as they reduce errors. The document frequency metric was also used for optimal feature selection, with Naïve Bayes. The NB classifier runs on the basis of probabilistic classifiers based on prior assumption that features present in the search space are independent of one another. NB when combined with Bernoulli event model shows impressive results compared to other counterparts as stated by [18]. Another work on classifier studied about AkNN model with TF/IDF and cosine similarity measures. It was found that the model outperformed conventional model in both clustering and classification [6]. Another similar work in this direction found that, SVM showed better performance than the K* and kNN-IBK in detecting fake news and sentiment analysis [5]. The two works such as [8], [10] & [3] compared SVM, Naive Bayes and kNN algorithms to identify their weaknesses and showed how the number of attributes of the every feature contribute in the performance of the algorithms using accuracy metrics. The same was done in [12] to test the effectiveness of Rule Induction, Decision trees, Artificial Neural Networks, NB and kNN algorithms. In the high dimensional data is used for text classification with large number of categories. And [7] provided a survey about theory and techniques of document classification. The proposed work is a comparative analysis of SVM, Decision Tree, J48, IBK, kNN and Naïve Bayes algorithms to study the strengths and weaknesses of the same. The findings from the study may be helpful to adapt changes such as type of learning, feature selection optimization and search space minimization.

3. METHODOLOGY

The basic working methodology of each algorithm will be discussed in the following sections.

I. Naïve Bayes(NB)

It is a pattern classifier based on class restricted likelihood using Bayes' formula. The Naïve hypothesis followed by the model is self-determining. But this procedure will be disturbed in the NLP [6] setting where the conversational framework calls for varied requirements in terms of pragmatics, syntactic and document semantic knowledge extraction. In consequence of this ideology, the conditional independence assumption can be used for likelihood p (a/c) to identify the category. The correct category can be identified using the equation (1) & (2)

$$p(a1, a2,... an/c) = p(a1/c) \cdot p(a2/c) \cdot ... \cdot p(an/c) \quad ......(1)$$

and,

$$c_{NB} = \text{argmax}_{c \in C}\ p(c) \quad a \in A\ p(a/c)......(2)$$

NB classifier considers word positions by undertaking a walk over the index through each word position in the given text document based on the following equation (3)

Positions←all word positions in test text document.

$$C_{NB}= \text{argmax}p© \quad q \in \text{position}\ p(wq /c)......(3)$$

The computations for the algorithm were carried out in log space to avoid the underflow problem using the equation (4) as it would also reduce time complexity. The major benefit of this algorithm is that it can learn from limited attributes, their mean, dependence and variances among variables to deliver effective results.

$$c_{NB} =\text{argmax}\ c \in C\ \log(x) + \quad q \in \text{positions}\ \log(wq /c)......(4)$$

The features in the log space gives the linear function of input features to predict the category of the sample so the classifier is popularly called linear classifier.

II. Support Vector Machine

It is a binary classifier to classify samples between one of the two available classes [5]. The large margin is obtained through the vector space to demarcate the decision boundary between the two classes that is maximum distance from the data points in the training data. The nearest such distance is called the edge of the classifier and the points are called support vectors. The decision margins are wider for good classifier models. The documents are separated through linear separation and kernel functions. To achieve this the decision surface is called hyperplane denoted by, m.d = n, where d denotes the documents and vector t and constant b are learned from the training set. For those documents that cannot be linearly separated, kernel tricks are used in its feature space. The data in higher dimension feature space can be obtained using $\beta(x)$ as defined in equation (5).

$$F(x) = \text{sign}. (m. \beta\ (x) +n) ...... (5)$$

The kernel function maps the new points to the feature space using the hyperplane. The various kernels used in the procedure are called, Linear, Gaussian, Exponential, Polynomial, Hybrid Kernels etc. The accuracy of the classifier is dependent on the selection of appropriate kernel function. It can be determined based on the type of data and requirements for the tasks, though it is a case of time complexity, benefits of accurate classifiers is more.

III. Decision Tree

It is mainly a regression based classifier used for prediction tasks [15].The hierarchy of rules are formed similar to trees to build the classification model using the training data. The training data is initially classified and rules are determined and standard that matches the learning are applied. The internal nodes are called features and branches are called feature values and test is applied on the features to derive decision rules for future samples. The leaves depict the category or class or sense of the learning model. CART, C4.5, ID3 and J48 are various versions of decision trees. The decision trees are popular for their simple interpretation of results, learning over large feature space, though with time overhead and explicit decision making rules. However, this is limited by the terminal node rules with minimal training samples not yielding accurate estimations for class labels. DTs are also used for Information Retrieval by generating definitions for concepts.

J48 algorithm

It is a open source version of C4.5 algorithm based on the concept of information entropy. The data when subsetted into smaller group of samples can be used to train the samples for classification. It helps to include all the features for the study. The J48 is concerned about the standardization of the information gain when selecting a particular attribute for splitting the data. The decisions are made with the attributes that obtain highly standardized information gain values and the steps are repeated for other subsets. If every instance in the subset belong to the same category the splitting is stopped and leaf node is assigned a category by the decision tree. When no attributes obtain expected information gain, the value of node higher than the present level is assigned as probable value of the class. It applies for both continuous and discrete variables.

IV. K – Nearest Neighbour algorithm

It contrasts the given [15] test sample with training instances related to it. A nearest neighbour is obtained for every point in the data using the training sample the neighbours give the value of the unknown vectors. It consists of three principal components, first, that uses a labelled approach, second, a distance metric to compute the distance between the objects and third denotes the recent known value of k (neighbours). The distance of unknown object from the known object is calculated and nearest k neighbours are identified. The class labels are that of the neighbour's class labels. The distance is calculated using, Euclidean and other similar metrics, but the learning curve is large and also partially suitable for classification problems.

IBK algorithm

Instance Based K algorithm [5] is similar to kNN except for the fact that it is able to standardize the range of features, incremental occurrence of processes (learned knowledge) and rules for inclusion of missing values. The distance metric is similar to kNN but search technique differs to find nearest neighbours. The memory about training samples is effective in classification of new samples and the algorithm uses distance metrics to predict new classes thereby reducing the training curve. It however maintains a large storage space for memorizing training cases, that incurs high computational costs.

4. EXPERIMENTAL SETUP

The experiments were conducted using Waikato environment for Knowledge analysis (WEKA) tool to compare these algorithms on various datasets [19]. It is an open source tool for virtualization of data analytics procedures such as clustering, data pre-processing, regression, visualization, classification and feature selection. The figure 1 states the process flow carried out in this proposed work.

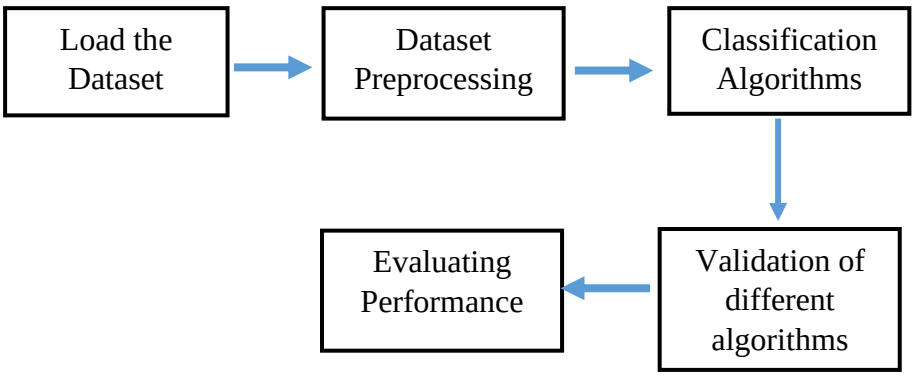

Fig. 1: Process of workflow

5. COMPARATIVE ANALYSIS OF ALGORITHMS

We compare four supervised classification algorithms for the proposed work such as, Naïve Bayes, SVM based (SMO algorithm), decision based (J48 algorithm) and KNN based (IBK algorithm). The algorithms are named with aliases given in the brackets as WEKA uses these names in its subroutines. The input data is fed in .arff format for both numerical and nominal attributes. The proposed work uses six datasets that are mixture of both attributes.

The datasets belong to various disciplines such as, Engineering, Social Science and Life Sciences. The datasets are obtained from WEKA repository [17] with different number of attributes. This is done to ascertain the ability of each algorithm in handling such instances. The dataset is divided into training and test sets to build and evaluate data models. The results are compared using the values obtained for the metrics such as, true positive, true negative, false positive and false negative. The values are plotted as confusion matrix as given in Table: 1 for better understanding. The positive and negative values are predictions made by the classifier for a particular observation sample.

Table 1: Confusion Matrix

	Predicted	
	Categorized Positive	Categorized Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

Where,
True Positive (TP): Classifier classifies positive class as positive.
False Positive (FP): Classifier classifies negative class as positive. Incorrect classification.
True Negative (TN): Classifier classifies negative class as negative.
False Negative (FN): Classifier classifies positive class as negative. Incorrect classification.
The confusion matrix in Table 2 is obtained as True positives (TPj) are instances accurately classified into a specific positive class, True Negatives (TNj) are instances correctly classified to specific negative class divided by the sum of TP, TN and FP, FN. It is denoted by the equation (6)

$$Accj = \frac{TPj + TNj}{TPj + TNj + FPj + FNj} \quad \ldots\ldots\ldots(6)$$

Recall score gives the number of samples retrieved by the model that are relevant to the user query or relevant to the task. Denoted by equation (7).

$$Rj = \frac{TPjj}{TPj + FNj} \quad (7)$$

Precision score gives the ratio of samples accurately assigned to its class. Denoted by equation (8).

$$Pj = \frac{TPi}{TPj + FPj} \quad (8)$$

F1 measure is the combination of the above metrics, denoted by equation (9).

$$F1 - measure = \frac{2 \times Pj \times Rj}{(Pj + Rj)} \quad (9)$$

where j denotes the instances of the dataset.

Dataset Description
Numeric Datasets – Iris dataset comprises of 150 instances of 5 different flowers, Glass dataset consist of 214 instances of ten types of Glass and Ionosphere consist of 351 instances of 35 types radar returns of the atmosphere. The data were randomly picked to avoid any bias in the algorithm performance.

Nominal Datasets – Vote data consist of 435 instances and 17 attributes related to voters in a particular electoral system and Soybean consists of 686 instances and 36 types of soybean disease diagnosis. The description is given in the table 2.

Table 2: Dataset Characteristics

Data sets	Total number of Instances	Total number of Attributes	Type
Vote	435	17	Nominal
Iris	150	5	numeric
Glass	214	10	numeric
Ionosphere	351	35	numeric
Soybean	683	36	Nominal

Accuracy
The accuracy obtained by the algorithms on the datasets mentioned is given in the Table -3 and subsequently in the Figure 2. From the Table-3 it is found that, Decision Tree and SVM performed well for Vote datasets with minimal difference in their performance. For Iris data, SVM and NB shown better performance, though not very different from other two algorithms. For Glass dataset all the algorithms have shown reduced performance though IBK is better to some extent.

In Ionosphere data, Decision Tree performed better than other algorithm and a thin performance improvement compared to SVM. Soybean dataset was modelled better with SVM where all other algorithms have still showed enough efficiency. The SVM and DTs are proven to be effective for both types of attributes as stated by the overall accuracy of the data.

Table 3: Comparison of accuracies of all datasets

	Naïve bayes	SVM (SMO)	K- nearest neighbor (IBK)	Decision Tree (J48)
vote	90.02	95.77	92.58	96.57

Iris	95.53	96.27	95.40	94.73
Glass	49.45	57.36	69.95	67.58
Ionosphere	82.17	88.07	87.10	89.74
Soybean	92.94	93.10	91.20	91.78

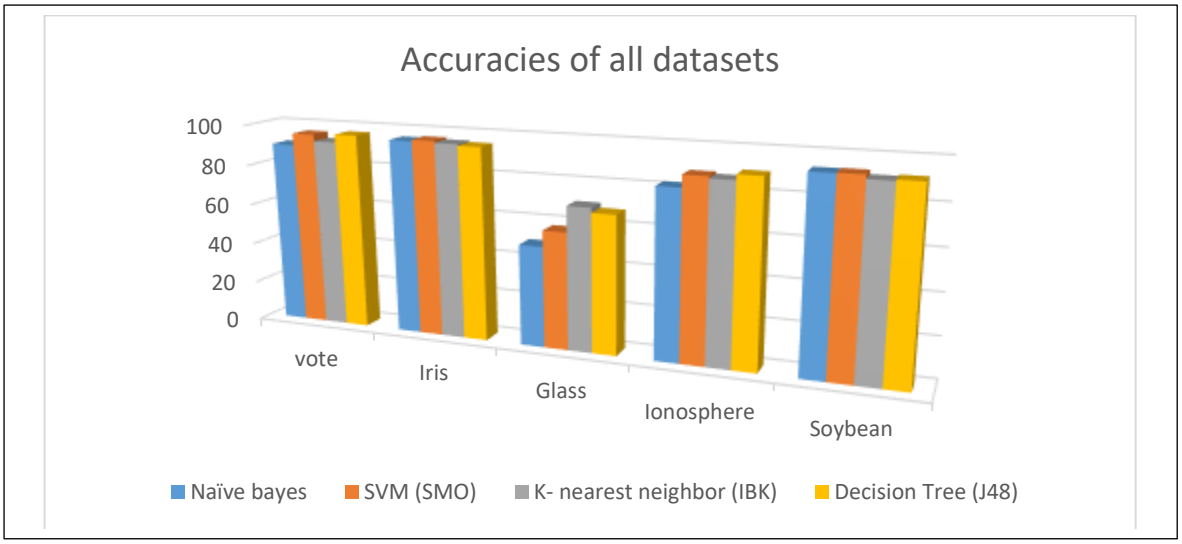

Fig. 2: Accuracies of all datasets

Precision

The precision obtained by the algorithms on the datasets mentioned is given in the Table -4 and subsequently in the Figure 3. From the Table-4 it is found that, Decision Tree and SVM performed well for Vote datasets with minimal difference in their performance compared to NB and kNN algorithms. For Iris data, all algorithm achieved a precision of 1.00 meaning, the dataset overfits the model. For Glass dataset all the algorithms have shown reduced performance though IBK is slightly better than DTs.

In Ionosphere data, SVM and IBK performed equally and better than other algorithms. Soybean dataset was modelled better with NB and SVM though it overfitted compared to IBK. The DTs and IBKs are proven to be effective for both types of attributes as stated by the overall precision of the data with minimal overfitting issues.

Table 4: Comparison of precision of all datasets

	Naïve bayes	SVM (SMO)	K- nearest neighbor (IBK)	Decision Tree (J48)
vote	0.95	0.98	0.96	0.97
Iris	1.00	1.00	1.00	1.00
Glass	0.48	0.63	0.71	0.70
Ionosphere	0.72	0.94	0.94	0.90
Soybean	1.00	1.00	0.95	0.96

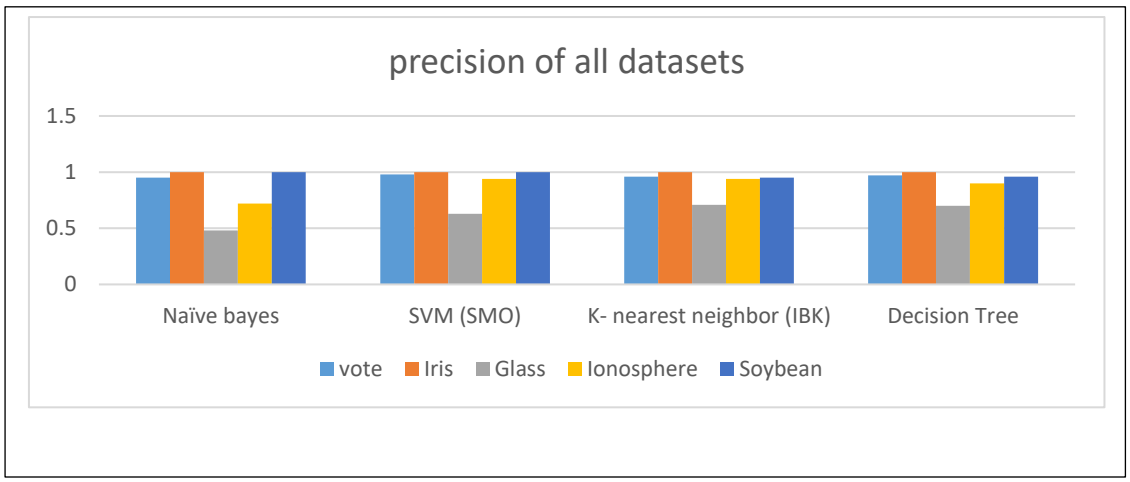

Fig. 3: Precision of all datasets

Recall

The recall obtained by the algorithms on the datasets mentioned is given in the Table -5 and subsequently in the Figure 4. From the Table-5, it is found that, Decision Tree and SVM performed well for Vote datasets with minimal difference in their performance. For Iris data, all three algorithms except DT overfitted the data. For Glass dataset all the algorithms have shown reduced performance though IBK is better to some extent.

In Ionosphere data, NB performed better than other algorithms and a considerable performance improvement compared to DTs. Soybean dataset was modelled better with DTs where all other algorithms have overfitted. The DTs are proven to be effective for both types of attributes as stated by the overall recall of the data.

Table 5: Comparison of recall for all datasets

	Naïve bayes	SVM (SMO)	K- nearest neighbor (IBK)	Decision Tree (J48)
vote	0.89	0.96	0.92	0.97
Iris	1.00	1.00	1.00	0.98
Glass	0.74	0.48	0.75	0.71
Ionosphere	0.86	0.72	0.69	0.82
Soybean	1.00	1.00	1.00	0.96

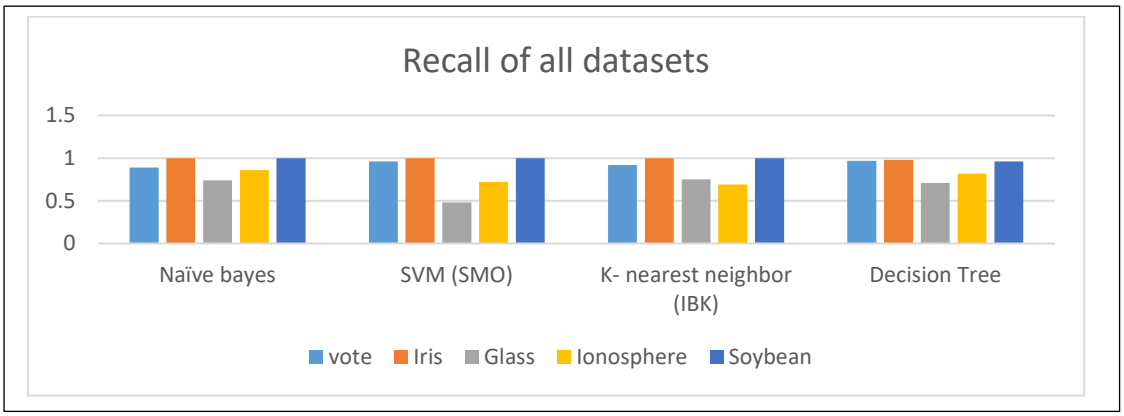

Fig. 4: Recall of all datasets

ROC

The Receiver Operating Characteristic (ROC) curve obtained by the algorithms on the datasets mentioned is given in the Table -6 and subsequently in the Figure 5. From the Table-6 it is found that, Decision Tree, IBK SVM performed well for Vote datasets with minimal difference in their performance. For Iris data, DT shown considerable performance, whereas all other algorithms over fitted the data. For Glass dataset all the algorithms have shown reduced performance though IBK is better to some extent.

In Ionosphere data, Naïve Bayes performed better than other algorithms with a considerable margin. Soybean dataset was modelled better with DT where all other algorithms over fitted the data. The DTs are proven to be effective for both types of attributes as stated by the overall Area under ROC of the data without overfitting.

Table 6: Comparison of Area _under_ ROC of all datasets

	Naïve bayes	SVM (SMO)	K- nearest neighbor (IBK)	Decision Tree (J48)
vote	0.97	0.96	0.96	0.98
Iris	1.00	1.00	1.00	0.99
Glass	0.73	0.78	0.80	0.79
Ionosphere	0.94	0.84	0.83	0.89
Soybean	1.00	1.00	1.00	0.98

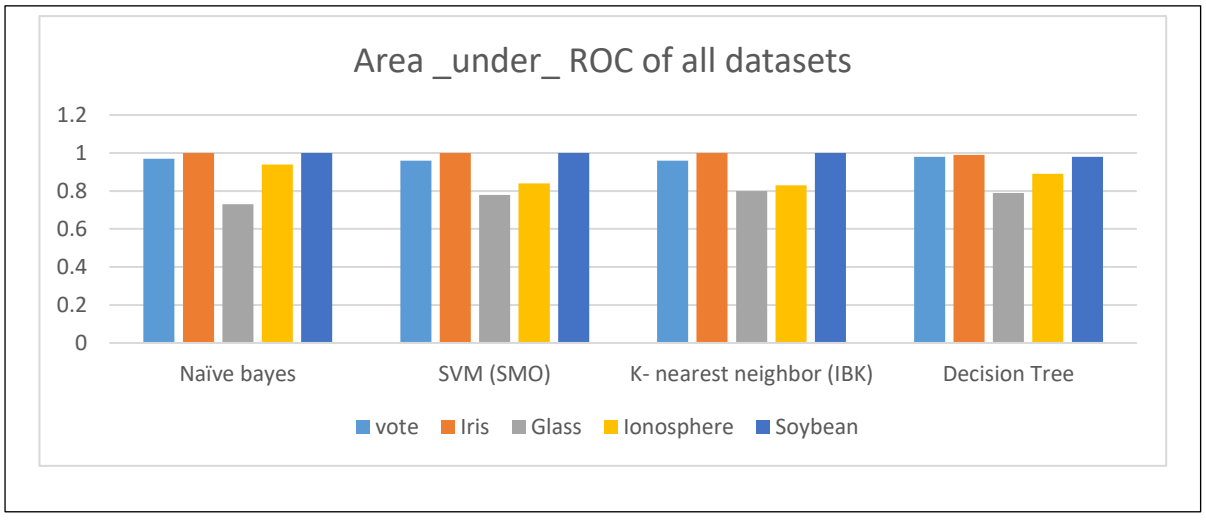

Fig. 5: Area _under_ ROC of all datasets

F1 measure

The accuracy obtained by the algorithms on the datasets mentioned is given in the Table -7 and subsequently in the Figure 6. From the Table-7 I it is found that, Decision Tree and SVM performed well for Vote datasets with no difference in their performance. For Iris data, DTs has shown better performance while other algorithms over fitted. For Glass dataset all the algorithms have shown reduced performance though IBK is better to some extent.

In Ionosphere data, Decision Tree performed better than other algorithm and a considerable performance improvement compared to SVM. Soybean dataset was modelled better with IBK algorithm while NB and SVM over fitted the data. The IBK and DTs are proven to be effective for both types of attributes as stated by the overall F1 measure of the data.

Table 7: F-measure of all datasets

	Naïve bayes	SVM (SMO)	K- nearest neighbor (IBK)	Decision Tree (J48)
vote	0.92	0.97	0.94	0.97
Iris	1.00	1.00	1.00	0.99
Glass	0.58	0.53	0.72	0.70
Ionosphere	0.78	0.81	0.79	0.85
Soybean	1.00	1.00	0.97	0.95

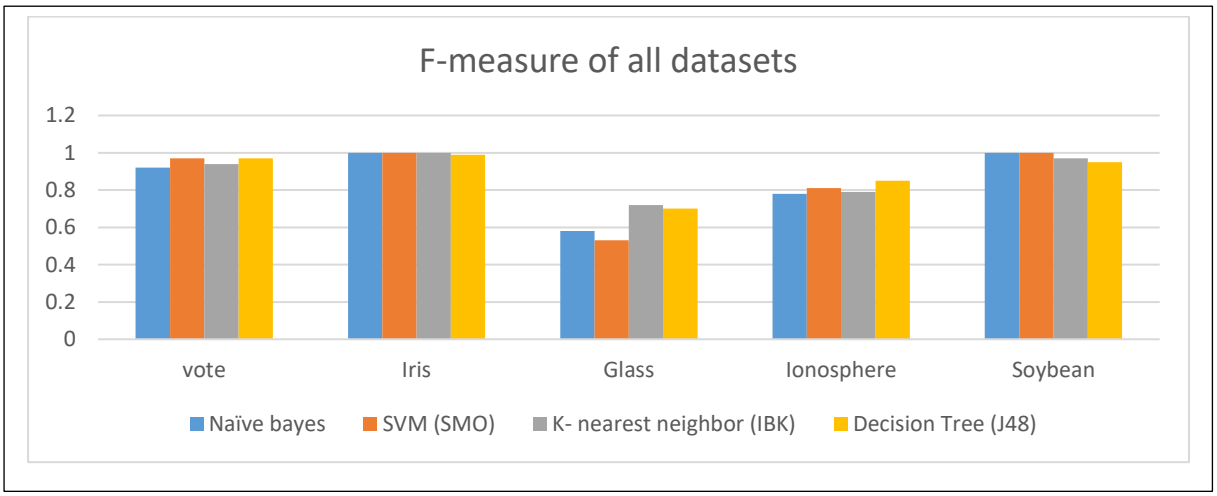

Fig. 6: F-measure of all datasets

6. RESULTS AND DISCUSSION

The aim of the study is to study the strengths and weakness of various ML algorithms for various data models. All the algorithms taken for the study were determined based on the learning type, types of attributes and search space optimization. From the experimental results, it was found that, six algorithms have shown a better performance in one or other metric and it was difficult to determine one algorithm that performed well for all kinds of data. The decision trees were found to be suitable for analysis as its performance is efficient in four of the five metrics taken for the study. In terms of accuracy, DTs and SVM come closer to each other for the datasets. The interesting findings happened in the recall metric, where, NB and IBK shown remarkable performance when compared to other algorithms. Also Iris dataset is the most over fitted according to the various metrics, this is mostly due to the highly balanced class representation which does not happen in real world. IBK can be used for tasks involving moderately represented classes. And NB can be used for problems that require higher recall performance.

7. CONCLUSION AND FUTURE WORK

The role of automatic text classification is studied with the help of the experimental study undertaken in the proposed work. The knowledge obtained by the models can be used for future classification tasks in similar datasets in the respective domains. The algorithms that achieved better accuracy as such as DTs and SVM can be applied for tasks where accuracy is the significant aspect compared to time complexity. IBK can be used for tasks involving moderately represented classes. And NB can be used for problems that require higher recall performance. From the study, it was also found that the dataset taken for analysis should also be applied with Exploratory Data Analysis to ensure no bias occurs. The performance metrics taken for this study is enough to ascertain the ability of each algorithm for a particular study. The proposed work can be helpful for the groundwork for other researchers in this domain. Applications of these algorithms to document classification and information extraction can be explored in future with better feature selection techniques.

8. REFERENCES

[1] Aggarwal, C. C., & Zhai, C. (2012). A survey of text classification algorithms. In Mining text data (pp. 163-222). Springer, Boston, MA.

[2] Aliwy, A. H., & Ameer, E. A. (2017). Comparative study of five text classification algorithms with their improvements. International Journal of Applied Engineering Research, 12(14), 4309-4319.

[3] Colas, F., & Brazdil, P. (2006, August). Comparison of SVM and some older classification algorithms in text classification tasks. In IFIP International Conference on Artificial Intelligence in Theory and Practice (pp. 169-178). Springer, Boston, MA.

[4] Dharmadhikari, S. C., Ingle, M., & Kulkarni, P. (2011). Empirical studies on machine learning based text classification algorithms. Advanced Computing, 2(6), 161.
[5] Dwivedi, S. K., & Arya, C. (2016, March). Automatic text classification in information retrieval: A survey. In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (pp. 1-6).
[6] Kadhim, A. I. (2019). Survey on supervised machine learning techniques for automatic text classification. Artificial Intelligence Review, 52(1), 273-292.
[7] Khan, A., Baharudin, B., Lee, L. H., & Khan, K. (2010). A review of machine learning algorithms for text-documents classification. Journal of advances in information technology, 1(1), 4-20.
[8] Korde, V., & Mahender, C. N. (2012). Text classification and classifiers: A survey. International Journal of Artificial Intelligence & Applications, 3(2), 85.
[9] Kumbhar, P., & Mali, M. (2016). A survey on feature selection techniques and classification algorithms for efficient text classification. International Journal of Science and Research, 5(5), 9.
[10] Mamoun, R., & Ahmed, M. A. (2014). A Comparative Study on Different Types of Approaches to the Arabic text classification. In Proceedings of the 1st International Conference of Recent Trends in Information and (Vol. 2, No. 3).
[11] Mishu, S. Z., & Rafiuddin, S. M. (2016, December). Performance analysis of supervised machine learning algorithms for text classification. In 2016 19th International Conference on Computer and Information Technology (ICCIT) (pp. 409-413). IEEE.
[12] Pawar, P. Y., & Gawande, S. H. (2012). A comparative study on different types of approaches to text categorization. International Journal of Machine Learning and Computing, 2(4), 423.
[13] Shah, K., Patel, H., Sanghvi, D., & Shah, M. (2020). A comparative analysis of logistic regression, random forest and KNN models for the text classification. Augmented Human Research, 5(1), 1-16.
[14] Tang, B., He, H., Baggenstoss, P. M., & Kay, S. (2016). A Bayesian classification approach using class-specific features for text categorization. IEEE Transactions on Knowledge and Data Engineering, 28(6), 1602-1606.
[15] Thangaraj, M., & Sivakami, M. (2018). Text classification techniques: A literature review. Interdisciplinary Journal of Information, Knowledge & Management, 13.
[16] Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. Journal of machine learning research, 2(Nov), 45-66.
[17] UCI Machine Learning Repository: http://archive.ics.uci.edu/ml/datasets.html.
[18] Vala, M., & Gandhi, J. (2015). Survey of text classification technique and compare classifier. International Journal of Innovative Research in Computer and Communication Engineering, 3(11), 10809-10813.
[19] Weka: http://www.cs.waikato.ac.nz/~ml/weka/

