**Received:** 27 March 2020; **Revised:** 08 June 2020; **Accepted:** 09 June 2020; **Published Online:** 10 June 2020
*Researchpedia Journal of Computing, Volume 1, Issue 1, Article 3, Pages 22-29, June 2020*

*Digital Object Identifier 10.1111/RpJC.2020.DOI Number*

# Evaluation of State-of-the-Art Classifiers: A Comparative Study

**Abdulwahab Ali Almazroi[1], Osama Abdo Mohamed [1,2], Azra Shamim[1], Muhammad Ahsan Qureshi[1,*]**
[1]College of Computing and Information Technology, University of Jeddah, Khulais
[2] Faculty of Science, Math. Department, Zagazig University, Egypt

Corresponding author: Muhammad Ahsan Qureshi (e-mail: maqureshi@uj.edu.sa).

**ABSTRACT** This study presents a comprehensive survey on widely used classifiers: Support Vector Machine, Random Forest, Naive Bayes, Stochastic Gradient Descent, and K-nearest Neighbor. These algorithms cover the research areas such as image processing, text classification, businesses, medical. A description of each algorithm is described along with the working of each algorithm in this study. Further, the strengths and weaknesses of each classifier are highlighted. Moreover, the application areas of each classifier are spotlighted. The evaluation metrics for the classifiers are also discussed. This research guides researchers in the selection of appropriate classifiers for their problems.

**Keywords** Classifiers; Support Vector Machine; Random Forest; Naive Bayes

## I. INTRODUCTION

Classifiers are one of the most common prediction tools in the area of data mining. The goal of classifiers is to predict the outcome based on a given input. A classifier constructs a prediction model for the prediction of the class of an object by making a connection between the attributes of input cases and then applied this model to predict about new cases [1]. Specifically, classification learns a target function f (prediction model) that maps each attribute set X to one of the predefined class labels y [2]. The input cases are divided into two parts called training dataset and testing dataset [3]. A classification process is divided into two steps: training and testing steps [4]. In the training step, the training dataset is used to make a prediction/classification model based on the attributes of input cases. In the training set, both the input attributes and prediction value (class labels) are presented. On the other hand, the testing dataset consists of only input attributes without class labels. The testing dataset is used to predict the class labels of new cases by using their input attributes. Further, the performance of a prediction model based on different performance measures such as accuracy, precision, recall, F-score, execution time, and error rate is evaluated on the testing dataset. The mathematical representation of a classification problem is:

Given a training dataset $T_n$. Let $T_n$ consists of a set of n cases; $T_n = \{C_1, C_2, C_3 \ldots C_n\}$, where n is the total number of the cases in $T_n$. Each case of $T_n$ contains a set of attributes where one attribute is the class labels. Let $A = \{a_1, a_2, a_3, \ldots. A_k\}$ is the set of attributes and $L_m = \{L_1, L_2, L_3, \ldots L_m\}$ is a set of m class labels that differentiates the classes of $T_n$ where m represents the total number of classes in $T_n$. In case of binary classification, m = 2; similarly in multi classification, m > 2. $\forall$ Ci $\in$ Tn $\exists$ A | {ak $\in$ A} = $\exists$ {Lc $\in$ Lm}. The objective of a classifier is to assign an accurate class label for each new case by building a model for class attribute as a function of other input attributes.

In machine learning history, different classification algorithms are proposed in the literature of data mining that cover a diverse set of problems including Stochastic Gradient Descent, K Nearest Neighbors, Support Vector Machine, Naive Bayes, Random Forest, Extra Trees, Decision Trees, and Neural Networks. These classifiers are applied in diverse classification tasks like image processing, email spam, credit card transactions, tumor cell, and semantic analysis. However, the strengths and weaknesses of each algorithm are different from each other. Some algorithms outperform others in specific classification problems. The applicability of these algorithms also depends on different parameters such as training time, parameter tuning, and prediction time. For instance, some algorithms perform better on large datasets while others provide better result on small datasets. Therefore, the selection of appropriate classifiers is of vital importance in critical domains. Mostly, this selection is based on the known behavior of the classifiers. However, for the generalization and replication of results a systematic evaluation of classifiers is required [5].

This paper presents a comprehensive survey of state-of-the-art classifiers. Specifically, five prominent classifiers, namely, Random Forest, Support Vector Machine (SVM), Naive Bayes, Stochastic Gradient Descent (SGD), and K Nearest Neighbors (KNN) are targeted in this study. A description of each selected classifier is provided along with a discussion of working classifiers. Further, the strengths and weaknesses of the selected classifiers are provided. Moreover, the specific application areas of the classifiers are also highlighted in this study. This study also reports the performance evaluation measures for classifiers.

The rest of the paper is organized as follows. Section 2 describes the Support Vector Machine, Random Forest, Naive Bayes, Stochastic Gradient Descent, and K-Nearest Neighbors classifiers. Section 3 highlights the advantages and disadvantages of these algorithms. Section 4 describes the application areas of selected classifiers. Section 5 presents the performance evaluation measures of classifiers. Section 6 concludes this work.

## II. Literature Review

This section presents a review of five important classifiers, namely, Support Vector Machine, Random Forest, Naive Bayes, Stochastic Gradient Descent, and K-nearest neighbor that are targeted in this study.

### *A. Support Vector Machines*

SVMs are accurate and most popular classifier introduced in 1992 by [6]. SVMs are now playing an important and active role in the field of machine learning research by providing accurate and robust methods of classification [7]. A SVM is a binary linear classifier based on a nonprobabilistic approach that is a key strength of SVM [8]. SVM is a classifier based on machine learning theory to maximize predictive accuracy and avoid the over-fit problem of data automatically [9]. The goal of a SVM is to separate data across a plane (decision boundary) described by feature vectors that are a subset of the data. The subset of data that supports the plane is referred to support vectors. The SVM splits all data points into two classes.

Assume a given set S of points xi $\in R_n$ with i = 1, 2,..., n. Each point $x_i$ in S belongs to either of the two classes and assigns a label $L_i \in \{-1, 1\}$. The data objects must have $\{x_1 \ldots x_n\}$ features and a class label ($y_i$). Specifically, a data object that is characterized by its feature vector belongs either of two classes: class 1 having label $L_i = 1$ and class 2 having label $L_i = -1$. The data is defined as presented in the Equation i **Error! Reference source not found.** below:

$$Data = \{(x_i, L_i) | \, x_i \in R^p, L_i \in (-1,1)\} \qquad 1 \geq i \geq n \qquad \text{(i)}$$

### *B. Random Forest Classifier*

Random Forests was developed by [10] in 1999. RF is the result of substantial modifications to bagging. Specifically, it is based on ensemble learning method called bagging for classification that creates a large collection of de-correlated decision trees [11]. Therefore, the performance of RF is very similar to bagging for many classification problems. Basically, it composed of a multitude of decision trees depends on the values of a random vector that are sampled independently having the same distribution for all decision trees [10]. Each tree generates a prediction for a specific input case and then the predictions generated by all trees are combined using either by voting or averaging. More specifically, RF classifier comprising of a collection of decision tree predictor {h(x,Θk ), k=1, ...} where the {Θk} are identically distributed and independent random vectors [10]. All trees cast a vote for the most popular class at a certain input x. A random vector for the kth tree is generated independently from the previous random vectors1,...,k−1, however, the distribution of the random vector is the same as of previous random vectors 1,..., k−1. The kth tree is then grown using the vector k and training dataset resulting in a classifier h(x, k). For the classifier, x is an input vector. After generating a collection of trees, each tree votes the most popular class.

At the first step, RF grows a forest of many decision trees in which every tree is created from an independent bootstrap sample from the training data. At each node, m variables are selected randomly out of all possible variables. Then, the best split is identified on the selected m variables to grow the trees to maximum depth. At last, vote or average of these trees is utilized for new predictions. The primary building block of a RF is a CART decision tree [13]. However, RFs solve the problem of over fitting data of decision trees. Similar to SVM, RF considered to be the most accurate method of classification [5]. It is used widely for the classification problems having a large dataset [12].

### *C. Naive Bayes classifiers*

NB classifiers are linear statistical classifiers and are considered as simple and very efficient [14]. These classifiers use a probabilistic model based on Bayes' theorem proposed by Thomas Bayes [28]. The theorem is presented below in Equation ii:

$$p(c_j \,|d) = \frac{p(d \,|c_j)\, p(c_j)}{p(d)} \qquad \text{(ii)}$$

$p(c_j \mid d)$ is case d probability in class $c_j$
$p(d|cj)$ is generating case d probability given class $c_j$
$p(cj)$ is the occurrence of class $c_j$ probability
$p(d)$ is the occurrence of case d probability.

BN classifiers chooses the most likely classification $V_{nb}$ from the given attributes $a_1$, $a_2$, $a_3$, …$a_n$ using the following formula presented in Equation iii [29]:

$$V_{nb} = argmax_{v_j \epsilon V} P(v_j) \Pi P(a_i|v_j) \quad \text{(iii)}$$

Where $P(a_i \mid v_j) = \frac{n_c + mp}{n+m}$

BN classifiers assume that the features of the dataset are independent mutually and this assumption is called class condition independence. However, this assumption is often violated in practice and still the performance of these classifiers is very well under this assumption. Often the class independence condition is violated in real-world problems; however, NB classifiers show good perform [15]. BN classifiers can outclass other alternatives on small datasets [16]. These classifiers are known as easy to implement, relatively robust, accurate and fast classifiers [14]. In BN, a Bayesian network is composed of an acyclic directed graph. Each node of the graph has a probability distribution [4]. In the graph, nodes encode variables whereas edges describe the conditional probability. The density of the edges presents the complexity of the network where sparse and dense networks show simple and complex probabilistic modeling, respectively. BN classifiers are well-known and most successful classifiers for the classification of text documents. These classifiers are most suitable to the inputs having high dimensionality value [14].

### *D. Stochastic Gradient Descent Classifiers*

SGD classifiers are popular in deep learning research to solve the optimization problem and became an important optimization method [18]. It is a modified version of the standard gradient descent algorithm. Gradient descent method is used in SGD classifiers to find a minimum of a function by identifying the direction (in parameters' space) in which the slope of the function is the rising most suddenly and follow a direction opposite to this slope [19]. Therefore, SGD classifiers minimize the loss function by calculating the gradient for each training example and nudge parameter space in the right direction, that is, the opposite direction of the gradient. In this way, it minimizes the inaccuracy of predictions. The following equation iv is used to estimates the gradient for each iteration based on randomly picked single example $z_t$ [30]:

$$\omega_{t+1} = \omega_t - \gamma_t \nabla_\omega Q(z_t, \omega_t) \quad \text{(iv)}$$

In SGD the value of gradient $\partial E/\partial wki$ is approximated by the gradient for a point $\partial En/\partial wki$ and is achieved by summing over the whole training dataset [21]. It finds the find the coefficient of the cost function with the associated condition that minimizes the cost margin. It performs experiments with different coefficient values and cost function to estimate the cost for each case of training dataset [3]. The predicted cost values are compared with actual values to select the coefficient and cost function that results in the lowest cost value. Then it updates the value of the coefficient. In SGD classifiers, the value of gradient is computed by summing the entire training dataset and updating the weights after each example of the training dataset instead of after the whole training dataset. The performance of these classifiers is amazing on large scale and sparse machine learning problems [20].

### *E. Nearest Neighbor Classifiers*

Nearest neighbor classifiers are simple, non-parametric, and memory based classifiers [22][23]. These classifiers users' instance-based learning [26]. The working principle behind these classifiers is to identify a number of cases from training dataset that are closest to the testing dataset in distance and then these closet cases are used to predict the value of new cases [24]. Commonly, the standard Euclidean distance is used to find the closet cases; however, other distance functions can be utilized to compute the distance. The Euclidean distance is calculated using following Equation v [2]:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad \text{(v)}$$

After computing the distances, it keeps the k (is a fixed integer and must be K≥1) closest training cases. The closet training cases are ordered using distance measure such that the nearest case is the closest case to new case and the second nearest neighbor is the second close case and so on. Then the most common class label for these k-nearest neighbors is the predict class

label of the new case. Therefore, the nearest-neighbor method is a function of type $(X \times Y)\ n \times X \rightarrow Y$, and the distance function has type $X \times X \rightarrow R$ [25]. There are two main design considerations of these classifiers: selection of value of k and distance measure. The appropriate value of K provides optimal results.

### III. Advantages and Disadvantages of Classifiers

This section presents the advantages and disadvantages of the selected classifiers.

Table 1 provides a comparison of the classifiers in terms of advantages, disadvantages, and application areas. SVM and RF are known as the most accurate classifiers and handle non-linear data efficiently [5]. On the other hand, SVM and RF provide generalization capabilities by addressing the problem of over fitting data well. RF and SVM are considered as stable classifiers. BN and SVM are most suitable for high dimensional space problems [14]. On the other hand, KKN is not suitable for the problems having high dimensional space. RF and SGD performed well on large training datasets [20]; however, SVM is suitable for large datasets due to the long time required by SVM for training the classifier. Also, KNN is not suitable for large training data due to distance calculations. RF and NB, and KNN classifiers can handle missing values easily. RF has ability to handle to missing data automatically, whereas manual handling of missing data is required by KNN.

Feature scaling is not required in RF as compared to SVM, KNN, and SGD. The impact of noise is less in RF classifiers, while SVM classifiers show performance degradation on noisy data. KNN are able to handle noisy data well. NB classifiers require less data for training the model. The implementation of NB, KNN, and SGD classifiers is easy. SVM, KNN, and RF classifiers have high complexity due to which high resource and computing power are required. SGD and KNN require a number of input parameters and iterations. Among all the classifiers, NB classifiers are highly scalable classifiers.

**Table 1:** Advantages and disadvantages of Classifiers

| Classifiers | Advantages | Disadvantages |
|---|---|---|
| **SVM** | • It provides better accuracy as compared to other classifiers [6].<br>• It can handle nonlinear complex data easily and efficiently using Kernel trick [31].<br>• It has good generalization capabilities due to less over fitting problem [9].<br>• It has good stability and a small change of the data will not affect the performance of SVM significantly [32].<br>• It is effective in high dimensional spaces [31]. | • It has high algorithmic complexity [32].<br>• Feature scaling is required in SVM [32].<br>• Long time for training the classifier for large datasets [32].<br>• It is difficult to understand and interpret [32].<br>• It shows less performance on noisy datasets [33].<br>• To handle nonlinear data it is difficult to select appropriate kernel function [31].<br>• It has high memory requirement [32]. |
| **RF** | • It has good accuracy [5].<br>• It classifies large datasets efficiently [12] .<br>• It presents the importance of variables in classification process [10].<br>• It handles missing data automatically [34].<br>• It handles both categorical and continuous variables [35].<br>• Simple and fast for small and large problems [34].<br>• It can handle non-linear interactions [34].<br>• It performs automatic variable selection [34].<br>• No feature scaling is required [35]. It handles outliers<br>• It is stable [35].<br>• The impact of noise is less on it comparatively [35]. | • High Complexity [35]<br>• High Computational power requirements [35]<br>• High resources requirements [35]<br>• Longer Training Period [35]<br>• Less interpretable than a decision tree [34] |
| **BN** | • Fast, Simple and Efficient [14]<br>• Easy to implement [14]<br>• Very Scalable [39]<br>• Require less training data[37] | • Condition independence assumption is hardly true in real life applications [14].<br>• Data scarcity problem [39] .<br>• Chances of loss of accuracy [39]. |

| | | |
|---|---|---|
| | • It handles missing values [38].<br>• It provides best results for text classification problems [14]. | |
| **SGD** | • Faster than gradient descent and batch gradient descent [40]<br>• Easy to implement<br>• Suitable for large dataset [20]<br>• It performs updates more frequently [41].<br>• It can process more examples within the available computation time [42]. | • SGD requires a number of regulation parameters and iterations [42]<br>• It is also sensitive to feature scaling [42] |
| **KNN** | • Simple [22][23]<br>• No Training Period [43]<br>• Learn complex target functions [46]<br>• Easy to implement [43]<br>• Robust to noisy data [51]<br>• No data loss [46]<br>• | • High complexity [44]<br>• Easily fooled by irrelevant attributes [46]<br>• Requires the value of K [44]<br>• Requires the distance measure [44]<br>• High computation cost [51]<br>• Not suitable for large datasets [43]<br>• Not suitable to high dimensional problems [45]<br>• Requires feature scaling [43]<br>• Requires manual handling of missing values and outliers [43] .<br>• Sensitive to irrelevant features [51] |

## IV. Applications of Classifiers

The specific areas in which the classifiers provide good performance are also highlighted in this section.

The classifiers are applied into a wide range of fields such as medical, engineering, mathematical, agriculture, businesses, and computer science as show in Table 2. The common application area for all these classifiers is text classification. In the field of computer sciences, SVM classifiers are used in speaker recognition, Intrusion detection, text classification, and image processing for face detection, object recognition, and steganography. Other applications of SVM breast cancer detection, and protein structure prediction in the medical domain along with the engineering and mathematical problems. RF and NB classifiers are used for spam filtering. NB and KNN classifiers mostly used for recommender systems.

**Table 2:** Application Areas of Classifiers

| Classifiers | Application Areas |
|---|---|
| **SVM** | • Face detections in images [47]<br>• Protein Structure Prediction [48]<br>• Object Recognition [50]<br>• Handwriting Recognition [48]<br>• Intrusion Detection [50]<br>• Detecting Steganography in digital images [50]<br>• Mathematical and Engineering problems [50]<br>• Text classification [49]<br>• Image classification [49]<br>• Speaker identification<br>• |
| **RF** | • Text Classification [12]<br>• Network Intrusion Detection [12]<br>• Credit Card Fraud Detection [12]<br>• Email Spam Detection [12] |

| | |
|---|---|
| | • Gene Classification [12] |
| **BN** | • Text classification [39]<br>• Spam filtering [35]<br>• Recommender Systems [39]<br>• Semantic Analysis [39]<br>• Real-time Predictions [39]<br>• Medical Diagnosis [36] |
| **SGD** | • Text Classification [42]<br>• Natural Language Learning [42]<br>• Image Recognition [42] |
| **KNN** | • Text Classification [51]<br>• Medical Problems [51]<br>• Recommender Systems [51]<br>• Agriculture Problems [51] |

## V.

### Performance Evaluation of Classifiers

A set of parameters are used to report the performance of a classifier, namely, accuracy, precision, recall, F1-score, error rate [17]. The performance of a classifier is based on the numbers of records predicted correctly and incorrectly by the classifier. A confusion matrix is used to present the performance of a classifier. A confusion matrix is used to calculate the values of accuracy, precision, recall, F1-score, and error rate. An example of confusion matrix is presented below in Table 3:

**Table 3:** Confusion Matrix

| N=100 | Predictive Negative | Predictive Positive |
|---|---|---|
| Actual Negative | TN | FP |
| Actual Positive | FN | TP |

Where *TN* = True Negatives, *FN* = False Negatives, *FP* = False Positives, and *TP* = True Positives

#### *A. Accuracy*

The accuracy of classifiers reflects the predictive capabilities of a classifier [2]. The formula of the accuracy is given in Equation vi:

$$\text{Accuracy} = \frac{(\text{TP}+\text{TN})}{(\text{TP}+\text{TN}+\text{FP}+\text{FN})} \qquad \text{(vi)}$$

#### *B. Error Rate*

The error rate or misclassification rate is calculated using the Equation vii given below:

$$Error\ rate = 1 - Accuracy \qquad \text{(vii)}$$

#### *C. Precision*

The precision (p) of classification for each class is described by prediction accuracy for that class. The precision of the negative class is the number of correct negatives divided by the number of all negatives predicted for negative class. Precision is calculated using Equation viii.

$$p = \frac{TP}{TP+TF} \qquad \text{(viii)}$$

#### *D. Recall*

The recall (r) is the number of correct positive results divided by the number of positive results that should have been returned. The following Equation ix is used to calculate r.

$$\boldsymbol{r = \frac{TP}{P} = \frac{TP}{TP+FN}} \qquad \text{(ix)}$$

### *E. F1-Score*

The F1- score is the harmonic mean of precision and recall and described as a weighted average of the precision and recall (best is 1 and worst is 0) as shown in Equation x below:

$$\boldsymbol{F1Score = \frac{2pr}{P+R}} \qquad \text{(x)}$$

### *F. Specificity*

Specificity describes the proportion of correctly identified negative tuples and presents true negative rate as shown in Equation xi.

$$Specificity = \frac{TP}{P} \qquad \text{(xi)}$$

### *G. Sensitivity*

Sensitivity is the true positive rate that presents the proportion of correctly identified positive tuples as shown in Equation xii.

$$Sensitivity = \frac{TN}{N} \qquad \text{(xii)}$$

## VI. CONCLUSION

Currently, classifiers are playing an important role in machine learning research. The wide applications of state-of-the-art classifiers force the researchers to investigate the applicability, advantages, and disadvantages for the real-world problems. Therefore, this work presented a comprehensive survey on prominent classifiers: SVM, RF, SGD, and BN by highlighting their strengths and weaknesses. Also, the application areas of the classifiers (in which the performance of these classifiers is competitive) are spotlighted. The findings of this study show generality, suitability for non-linear data, accuracy, and stability of SVM and RF classifiers. Different metrics such as accuracy, precision, recall, F1-score, specificity, and error rate that are used to evaluate the performance of a classifier are also described in this study. It was inferred from the survey that most of the classifiers are utilized for the text classification problems. This work also provides guidelines for researchers and industry for the selection of the appropriate classifiers for the real-world classification problems.

## REFERENCES


[1] J. Ross Quinlan. C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[2] Han, J., Kamber, M. & Pei, J. (2012). Data mining concepts and techniques, third edition Morgan Kaufmann Publishers

[3] Shadi Diab, Optimizing Stochastic Gradient Descent in Text Classification Based on Fine-Tuning Hyper-Parameters Approach. A Case Study on Automatic Classification of Global Terrorist Attacks, International Journal of Computer Science and Information Security (IJCSIS), Vol. 16, No. 12, December 2018

[4] S.Neelamegam, E.Ramaraj, Classification algorithm in Data mining: An Overview, International Journal of P2P Network Trends and Technology (IJPTT) - Volume 3 Issue 5 September to October 2013

[5] Zhang C, Liu C, Zhang X, Almpanidis G (2017) An up-to-date comparison of state-of-the-art classification algorithms. Expert Systems with Applications 82: 128–150. Available: http://dx.doi.org/10.1016/j.eswa.2017.04.003.

[6] Cortes, C., Vapnik, V. Support-vector networks. Mach Learn 20, 273–297 (1995).

[7] X. Wu, V. Kumar, J. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. McLachlan, A. Ng, B. Liu, P. Yu, Z. Zhou, M. Steinbach, D. Hand, и D. Steinberg. Knowledge and Information Systems 14 (1): 1--37

[8] https://www.ugpti.org/smartse/resources/downloads/support-vector-machines.pdf

[9] Christopher J. C. Burges. 1998. A Tutorial on Support Vector Machines for Pattern Recognition. Data Min. Knowl. Discov. 2, 2 (June 1998), 121–167. DOI:https://doi.org/10.1023/A:1009715923555

[10] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[11] L. Breiman. Bagging predictors. 1994.

[12] Zakariah, M. (2014). Classification of large datasets using Random Forest Algorithm in various applications: Survey. International Journal of Engineering and Innovative Technology (IJJEIT), 4(3).

[13] https://gitee.com/it-ebooks/it-ebooks-2018-11to12/raw/master/RANDOMFORESTSFORBEGINNERS.pdf

[14] Raschka, Sebastian. "Naive bayes and text classification i-introduction and theory." arXiv preprint arXiv:1410.5329 (2014).

[15] Irina Rish. An empirical study of the naive bayes classifier. In IJCAI 2001 workshop on empirical methods in artificial intelligence, pages 41–46, 2001.

[16] Pedro Domingos and Michael Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. Machine learning, 29(2-3):103–130,1997.

[17] Powers, David M. W. 2007. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. SIE-07-001, School of Informatics and Engineering , Adelaide, Australia: Flinders University of South Australia, 24. http://www.flinders.edu.au/science_engineering/fms/SchoolCSEM/publications/tech_reps-research_artfcts/TRRA_2007.pdf

[18] E. YAZAN and M. F. Talu, "Comparison of the stochastic gradient descent based optimization techniques," 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, 2017, pp. 1-5, doi: 10.1109/IDAP.2017.8090299.

[19] Daniel Jurafsky and James H. Martin. 2000. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (1st. ed.). Prentice Hall PTR, USA.

[20] Islam, Md, Fazla Elahi Md Jubayer, and Syed Ikhtiar Ahmed. "A comparative study on different types of approaches to Bengali document categorization." arXiv preprint arXiv:1701.08694 (2017).

[21] https://www.inf.ed.ac.uk/teaching/courses/mlp/2016/mlp02-sln.pdf

[22] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning: data mining, inference and prediction. The Mathematical Intelligencer, 27(2):83–85, 2005.

[23] Louppe, Gilles. "Understanding random forests: From theory to practice." arXiv preprint arXiv:1407.7502 (2014).

[24] E. Fix and J. L. Hodges. Disciminatory analysis—nonparametric discrimination; consistency properties. Technical Report Project 21-49-004, Report No. 4, USAF School of Aviation Medicine, 1951

[25] Elkan, Charles. (2011). Nearest Neighbor Classification. 10.1007/978-0-387-39940-9_2920.

[26] Imandoust, S.B. & Bolandraftar, Mohammad. (2013). Application of K-nearest neighbor (KNN) approach for predicting economic events theoretical background. Int J Eng Res Appl. 3. 605-610.

[27] https://www.ugpti.org/smartse/resources/downloads/support-vector-machines.pdf

[28] Bayes, T. 1764. "An Essay Toward Solving a Problem in the Doctrine of Chances", Philosophical Transactions of the Royal Society of London 53, 370-418.

[29] Meisner, Eric. "Naive Bayes Classifier example." (2003). http://www.inf.u-szeged.hu/~ormandi/ai2/06-naiveBayes-example.pdf

[30] Bottou, Léon. (2010). Large-Scale Machine Learning with Stochastic Gradient Descent. Proc. of COMPSTAT. 10.1007/978-3-7908-2604-3_16.

[31] Auria, Laura, and Rouslan A. Moro. "Support vector machines (SVM) as a technique for solvency analysis." (2008). https://www.econstor.eu/bitstream/10419/27334/1/576821438.PDF

[32] http://theprofessionalspoint.blogspot.com/2019/03/advantages-and-disadvantages-of-svm.html

[33] https://medium.com/@dhiraj8899/top-4-advantages-and-disadvantages-of-support-vector-machine-or-svm-a3c06a2b107

[34] Cutler, Adele. "Random Forests for Regression and Classification." PowerPoint Presentation, Utah State University (2010).

[35] http://theprofessionalspoint.blogspot.com/2019/02/advantages-and-disadvantages-of-random.html
Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A bayesian approach to filtering junk e-mail. In Learning for Text Categorization: Papers from the 1998 workshop, volume 62, pages 98–105, 1998

[36] Joanna Kazmierska and Julian Malicki. Application of the na¨ıve Bayesian classifier to optimize treatment decisions. Radiotherapy and Oncology,86(2):211–216, 2008.

[37] http://theprofessionalspoint.blogspot.com/2019/03/advantages-and-disadvantages-of-naive.html

[38] https://towardsdatascience.com/the-naive-bayes-classifier-e92ea9f47523

[39] https://analyticsjobs.in/education/naive-bayes-classifier-in-machine-learning/

[40] Gower, Robert Mansel, et al. "SGD: General analysis and improved rates." arXiv preprint arXiv:1901.09401 (2019).

[41] https://medium.com/@divakar_239/stochastic-vs-batch-gradient-descent-8820568eada1

[42] https://analyticsindiamag.com/how-stochastic-gradient-descent-is-solving-optimisation-problems-in-deep-learning/

[43] http://theprofessionalspoint.blogspot.com/2019/02/advantages-and-disadvantages-of-knn.html

[44] https://people.revoledu.com/kardi/tutorial/KNN/Strength%20and%20Weakness.htm

[45] https://www.fromthegenesis.com/pros-and-cons-of-k-nearest-neighbors/

[46] Thampi, Sabu M., Alexander Gelbukh, and Jayanta Mukhopadhyay, eds. Advances in signal processing and intelligent recognition systems. Berlin: Springer International Publishing, 2014.

[47] Osuna, Edgar Elias. Support vector machines: Training and applications. Diss. Massachusetts Institute of Technology, 1998.

[48] https://data-flair.training/blogs/applications-of-svm/

[49] Nello Cristianini and John Shawe-Taylor, “An Introduction to Support Vector Machines and Other Kernel-based Learning Methods”, Cambridge University Press, 2000. Cervantes, Jair & García-Lamont, Farid & Rodríguez, Lisbeth

[50] Lopez-Chau, Asdrubal. (2020). A comprehensive survey on support vector machine classification: applications, challenges and trends. Neurocomputing. 10.1016/j.neucom.2019.10.118.

[51] Imandoust, Sadegh Bafandeh, and Mohammad Bolandraftar. "Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background." International Journal of Engineering Research and Applications 3.5 (2013): 605-610.