Journal of Engineering Sciences

Vol 11, Issue 3, March/2020
ISSN NO:0377-9254

Approach to efficiently choose Machine Learning Algorithms and Performance Measures

[1]ATHARVA TENDULKAR, [2]ATHASHREE VARTAK, [3]KASTURI DHOTRE,
[4]SHIVANI GUPTA
[1,2,3] B. E STUDENTS, [4]ASSISTANT PROFESSOR
THAKUR COLLEGE OF ENGINEERING AND TECHNOLOGY, MUMBAI.

Abstract— In Machine Learning, the primary issue faced isn't implementation of algorithm but choosing the algorithm to be implemented based on business requirement, hardware availability, time constraints, desired performance measure, data in hand, etc. The common approach followed is implementing all the feasible algorithms and then choosing the one that performs best. The main drawback of this approach is that a hefty amount of time is wasted for training and testing data on myriad of algorithms only to realize that few algorithms do not satisfy the required constraints. The proposed approach promotes understanding the desired outcome and the available data to choose the optimum algorithm that would maximize the performance measures. Also, the problem of choosing the appropriate performance measure is addressed. Instead of choosing random or all metrics for evaluating performance of model, based on business problem, best possible metrics could be found and should be optimized instead of focusing on all metrics.

Thus, this approach will help in saving significant amount of time as a single or few numbers of algorithms are chosen that fit the given constraints instead of trying out many algorithms and then determining which works the best. After selecting the necessary algorithm(s), the data will be needed to pre-processed and visualized to get insight about the data and modified in a way that is required for chosen algorithm(s). Lastly, model is evaluated based on only necessary metrics depending on the use case which would lead to better model suited for specific problem.

Keywords—Machine Learning, Classification, time-saving, Visualization, Model Evaluation, Logistic Regression, Decision Tree, Performance Measures, Naïve Bayes, KNN, SVM.

I. INTRODUCTION

Machine Learning is a field of computer science which gives the computer the ability to deduce patterns and learn without being programmed explicitly. Machine Learning's primary aim is to provide training to an algorithm based on previous data to perform required tasks which can be used for predictions. It is one of the most important technologies with applications like Speech recognition, recommendations, fraud detection, financial trading, etc. Today, Machine Learning is being used in almost all businesses in some or the other way to increase the profits and productivity. Machine Learning comprises of various algorithms which can be used for predictions based on the past data. Also, to check how well our built model is working, various performance metrics are available. It is necessary to choose appropriate algorithm and evaluation metrics for a problem else; it might lead to varied results which might not be very useful.

In this paper, we have shared our approach of how to select appropriate machine learning algorithm by taking some real-life machine learning problems. We have also emphasized on how to select appropriate performance metrics based on given problem.

II. RELATED WORK

In past, researches have worked extensively on various methodologies of choosing algorithms on the basis of data and the required outcome. Also, the past approach included using all the available performance measures instead of selecting only the particular ones required for that particular use case. Like, here, in [1] the authors have used a myriad of datasets and applied all the possible classifiers to those datasets. These classifiers also included all the possible variants of a particular classifier. These models were then evaluated on the basis of multiple generalized performance measures to determine if a particular classifier will perform better than a chosen base classifier. It infers that choosing the right machine learning algorithm is much more important than selecting a methodology and then choosing the algorithm. It also suggests that, sometimes, even the basic models perform as good as the models with sophisticated approaches perform. Thus, this can be further studied. But, in recent times far-reaching research has been carried out in these arenas. Now, considering the evaluation of performance measures, in [2], the author has described various metrics with

their description that we can use to evaluate our model. Author has also talked about few factors that we can consider while choosing right metrics. Also, emphasis is done on why it is important to choose right metrics. Only possible drawback is all these is explained without taking any real-life scenario which makes it difficult to understand when to choose which metrics while solving real problems. In addition to this, considering the various classifiers with respect to recommender systems, in survey [3], author has tried to answer most asked question about how to choose most appropriate algorithm for classification problem. Author has explained how various Data miner and Business dependent factors and technical factor play an important role in recommending most favorable machine learning algorithm for a given problem. Considering the decision tree classifier, in [4], the author has primarily explained the two basic splitting metrics - gini measure and information gain which cannot deal with imbalanced datasets. They have also introduced a new metric DKM which performs better in case of skewed data. This paper concentrated on the construction of decision of imbalanced dataset. Also, various techniques to convert the imbalanced dataset into balanced dataset are mentioned like SMOTE, sampling, etc. It also infers that C4.5 and CART perform the best for balanced dataset while DKN outperforms the mentioned approaches in case of skewed data. But this paper doesn't explain when and how the techniques to convert dataset into a balanced one should be used. It also doesn't specify any assumptions of various sampling techniques. Contributing to it, in [5], the authors signify the importance of sampling and give the cases where under sampling outperforms oversampling and vice versa. It also introduces a new sampling technique called as hybrid sampling. All these techniques are in consideration with the SVM classifier. Further in [6], the authors have explained in detail how SVM models can be built efficiently. It focuses on various issues faced while building models such as choosing right kernel functions, common mistakes made while scaling and appropriate feature selection. Finally, they have also discussed grid search method for appropriate parameter tuning. In [7], Authors have emphasized on working of Naïve Bayes along with its advantages and disadvantages. It also specifies various real time use cases where this algorithm fits well. In [8], author has tried to find out on what type of data does KNN performs best, for this, he had applied KNN on various types of dataset to reach to the conclusion that KNN works best when data attributes are nominal, but also produces significant result with numeric data. In [9] and [10], Authors have compared various and analyzed algorithms such as SVM, K-NN, Naïve Bayes, Decision tree on different dataset to find out which algorithms performs best depending on type of data.

III. METHODOLOGY

The standard approach that we have used to build machine learning models includes the following steps: Firstly, understanding of the business problem and the available data (along with all the constraints). Then, one or multiple algorithms are needed be chosen which can solve the problem efficiently. Once the algorithm is chosen, according to the requirements of the algorithms and the available tools, the data needs to be pre-processed so as to construct the model. Data pre-processing steps can vary from algorithm to algorithm as their basic mathematic basis differs. Then, appropriate model with appropriate parameters is applied on processed dataset and then the result is evaluated on the basis of a variety of performance measures dependent upon the expected outcome that needs to be solved.

The proposed method primarily includes detailed study of business problem that needs to be solved, data and the classification algorithms. It also includes understanding the mathematical basis of these algorithms so that it can help in process of optimization like while hyper-parameter tuning, etc. Also, this study will make the process of understanding and interpreting the results easy. The process of determining the algorithm to be chosen for the required dataset requires understanding of business constraints and data.

After detailed study of classification algorithms like Logistic Regression, Naïve Bayes, k-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, etc. comparison of all the algorithms on various factors such as performance, ability to handle new incoming data, computational time(testing and training), type of data pre-processing needed, nature of data, size of data , number of features etc have been made. Thus, depending on the given problem statement, suitable algorithm(s) are chosen taking into considerations above mentioned criteria. Once the algorithm is decided upon, data can be processed.

The transformation or pre-processing mentioned here can include dealing with missing values by replacing the values by mean, median or applying another prediction algorithm to determine the missing value. It may also include scaling the attributes, dealing with imbalanced datasets, feature selection, extraction, and so on. After doing required transformations, the algorithms should be applied upon the dataset. After training and testing the model, evaluation needs to be carried out on the basis of performance measures. There are different performance measures namely accuracy, f1-score, precision, recall, ROC curve etc. Therefore, each and every model is to be evaluated on the basis of the appropriate performance measures. After evaluating the models, they need to be optimized if

necessary or the desired results are not met. The process of optimization varies from algorithm to algorithm. Hyper-parameter tuning is one of the methods of optimization. For example, in KNN the optimum value of k can be found using elbow method to optimize the result or weights could be changed. On the other hand, in case of decision trees, the tree can be pruned to obtain the optimum results.

Here, in this approach, desired use-case have been found for each commonly used algorithm and tried to explain how any given problems can be approached by looking at business constraints and data. The following topics illustrate the above-mentioned process with respect to various classification algorithms.

A. Naïve Bayes

Naïve Bayes is the classification algorithm based on Bayes theorem (hence called is probabilistic classifier). It is based on naïve assumption that all the features are independent of each other which is not possible in real life. In spite of its naïve assumptions it is surprisingly found out through experience that it is suited for textual data. Even in documentation of sklearn it is mentioned that naïve Bayes is good estimator in case of textual data.

Dataset that was used was Twitter Dataset which contained around 50000 labelled tweets (racist or non-racist). Problem definition was to find if the given tweet is racist or not. Since dataset contained textual data, naïve Bayes was chosen as it tends to work well with textual data. Also, twitter data is being generated rapidly, so it is desirable that algorithm chosen must rapidly adapt to new data (train data quickly) and also prediction time should be quick. In Naïve Bayes, training time is less also predicting values is quick since it deals with probabilistic calculation. Hence, Naïve Bayes algorithm was chosen for this problem.

In this normal pre-processing step such as stock word removal, removing special characters, stemming and lemmatization were carried out to clean original text. The cleaned text was then converted to tfidf vector.

Fig. 1. Original data v/s Cleaned data

On this, Naïve Bayes algorithm was applied. In Naïve Bayes, there are not many parameters for tuning except alpha, hence hyperparameter tuning is not that important. But while converting to tfidf vector, tuned parameter such as min_df, max_df and max_features were tuned to get maximum output.

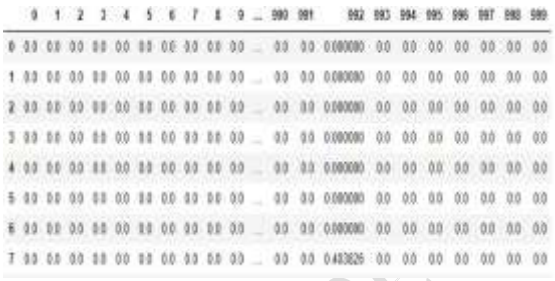


Fig. 2. TFIDF matrix

Since dataset is highly imbalanced (racist tweets hashtags are order of hundreds while non-racist tweets hashtags are order of thousands), accuracy cannot be used as a measure to determine performance. Also, for this problem, it is necessary to identify all the racist tweets correctly, as if tweet wrongly identified as racist in our application, then person can be bashed heavily on platform. Thus, here, important factor is being precise in our prediction of racist tweets, hence precision as performance measure was selected for this problem. Precision foe naïve Bayes algorithm was found to be 82.34%.

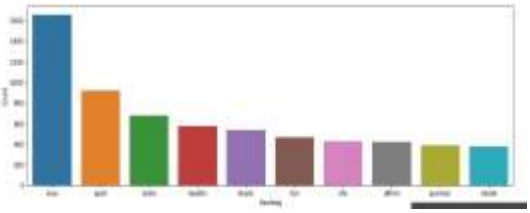
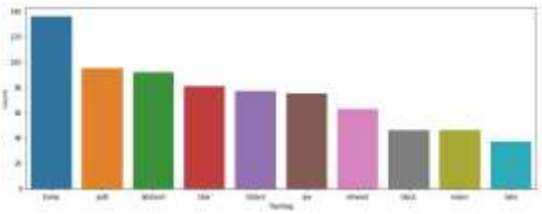


Fig. 3. Non-racist hashtag analysis



.Fig. 4. Racist hashtag analysis

Subsequently, other models were constructed by app lying the other algorithms into consideration. The pe rformances of these models can be summarized as fo llows:

Journal of Engineering Sciences

Table I. Performance of models on Twitter Dataset

Dataset	Algorithm	Performance Measure: **Precision**
Twitter Data set	Naïve Bayes	82.34 %
	k- Nearest Neighbours	71.67%
	Decision Tree	74.11 %
	Logistic Regression	81.43 %
	Support Vector Machine	83.17 %

Thus, from the observations in table I, it can be concluded though SVM gives little better precision than naïve Bayes, but training time required for SVM is much more. Incase of twitter data, our requirement is to train data quickly as data comes at high velocity and value of data degrades with time. Hence, in this case Naïve Bayes can be considered as better approach than SVM (since difference between them is not significant). Thus, it can be concluded that for this dataset, our assumption that Naïve Bayes will perform well is verified.

B. k-nearest Neighbors (KNN)

KNN is a classification algorithm based on the concept that similar things are always located close to each other. It is one of the simplest and easiest algorithms. KNN works by taking opinion of closest 'k' datapoints to predict the output (k is hyperparameter). KNN is also called as lazy learner because as model is not built during training phase which makes training phase faster. But while making predictions it does all the calculation of distance and sorting which makes prediction phase slower.

KNN model is constructed using Diabetes Dataset. In this problem was to find if the person has diabetes or not based on 8 independent features (attributes). Every day, many patients perform tests for diabetes. So main features should be that algorithm must be able to adapt to data and train it quickly, so prediction can be made based on latest data. Prediction may take time as patients are willing to wait if they get accurate results. In KNN, since it is a lazy learner, training phase takes almost no time and all computation is done while we are predicting value. This is the main reason of using KNN for this dataset. Also, since number of features is less, computation won't take very long time, so it's an added advantage. For pre-processing, only normalization of values was performed, since when dealing with distances, normalization becomes essential. Feature selection was not given that importance, as there were only 8 features and all of them were not very much dependent on each other.

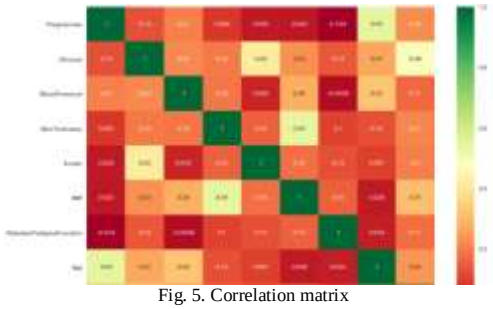


Fig. 5. Correlation matrix

For hyperparameter tuning, best value of k was chosen by iterating through values of k to see which value of k gives optimal results. We found out optimal value to be 21. Then KNN algorithm was applied on the dataset with right hyperparameters.

Fig. 6. Train v/s Test score (for choosing k)

For evaluating performance of the model, here, the primary objective was to determine if all the persons having diabetes were detected or not. So here, even if a person not having diabetes is classified as having diabetes is acceptable but not vice-versa. Thus, recall comes into picture. Recall is used to identify what proportion of people having diabetes was actually classified correctly. Recall (on person having diabetes) by this approach was recoreded as 76.56%.

Subsequently, other models were constructed by app lying the other algorithms into consideration. The pe rformances of these models can be summarized as fo llows:

Table II. Performance of models on Diabetes Dataset

Dataset	Algorithm	Performance Measure: **Recall**
Diabetes Dataset	Naïve Bayes	68.3 %
	k- Nearest Neighbours	76.56%
	Decision Tree	74.6%
	Logistic Regression	73.04 %
	Support Vector Machine	76.45 %

Thus, from the observations in table II, it can be con cluded that for this dataset, our assumption that KN

N will give perform well is verified. In this case, it p erforms as well as SVM with added advantage of le ss training time.

C. Decision Tree Classifier

Decision tree is one of the most widely used classification algorithm. The goal of this algorithm is to create a model that predicts the value of a dependent variable by learning simple decision rules inferred from the data features which are easy to visualize. Also, they are used when there can be multiple solutions to a particular problem that is there are multiple paths from the start point to the possible outcomes. Decision Trees are prevalent while dealing mainly with categorical data or a combination of numeric and categorical data. In addition to this, for every addendum to the training set, a new tree needs to be constructed which slows down the process if the data has high velocity. Hence, these can be used in cases where the data isn't frequently updated.

In the proposed model, the decision tree model is constructed using Nursery dataset which was developed to rank Nursery school applications in order to decide if admission can be offered. It consists of eight independent variables and one dependent variable having five outcome classes (recommended, priority, special priority, very recommended and recommend). The dataset is void of missing values. Here, as data is purely categorical and there are more than two outcome classes. Ergo, the information gain will be used as the splitting criteria. On visualization of the outcome classes, it can be inferred that the dataset is highly imbalanced due to the varying number of outcomes of the classes (Fig. 7).

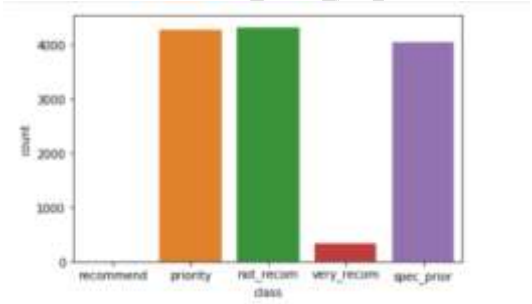


Fig. 7. Chart denoting imbalanced dataset.

For nursery dataset, accuracy is the crucial performance measure over others as we want our outcome to be correctly defined so as to make an accurate decision whether to admit the student or not. For determining the accuracy of a model, dataset needs to be balanced or else the outcome will be biased. The dataset can be balanced using various techniques like resampling, generating synthetic samples, etc. Here, as there are more than two outcome classes, Synthetic Minority Oversampling Technique, which creates synthetic samples, cannot be used. Also, under sampling majority class technique cannot be used as its majority of data will be lost owing to larger difference in the occurrence of various outcome variables. Instead, random oversampling minority class technique is used which adds copies of minority class to the dataset.

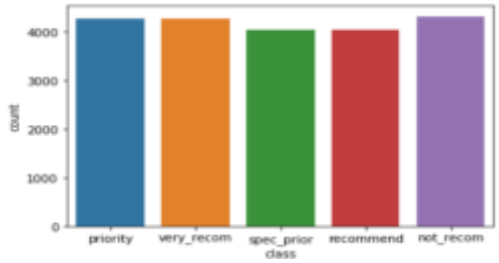


Fig.8. Balanced dataset after Random Oversampling

Overfitting is the only issue that can be incurred because of random oversampling as the same data is replicated. Thus, the data is first converted using dummies as sklearn library only processes numerical values. Thus, the number of features gets converted from 8 to 27 with possible values 0 or 1. On fitting the cross validated decision tree model, the training and testing accuracy is calculated for a range of values of depth to understand if the model is overfitting. We can say that the model is overfitted if the error on testing set is significantly greater than the error on training set.

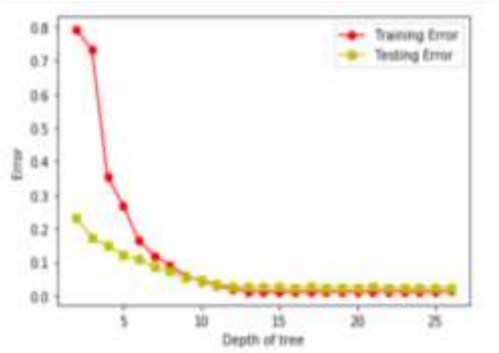


Fig. 9. Error v/s Depth mapping of training and testing set

From Fig. 9, it can be inferred that the model doesn't overfit as the error value is almost equal for all the values of depth. Here, as the dataset is balanced, accuracy needs to be considered as the crucial performance measure.

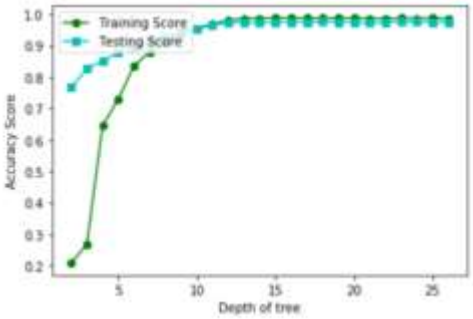


Fig. 10. Accuracy v/s Depth mapping of training and testing set

According to Fig. 10, the model performs extremely well at all the depth greater than depth = 10 and appr oaches almost perfect value of accuracy at greater de pths. Thus, the maximum accuracy obtained using D ecision Tree algorithm on Nursery dataset is 99.4% a t depth = 24.

On the other hand, when all the above stated algorith ms were applied on the said dataset, they didn't perf orm up to the mark as the decision tree classifier. Th e performances of these models can be summarized as follows:

Table III. Performance of models on Sonar Dataset

Datas et	Algorithm	Performance M easure: **Accura cy**
Nurse ry Dat aset	Naïve Bayes	83.74 %
	k- Nearest Neighbo urs	88.11 %
	Decision Tree	99.40 %
	Logistic Regression	91.41%
	Support Vector Ma chine	96.73 %

From the above observations from table III, obtaine d from the models of various algorithms, it can be i nferred that decision tree classifier performs the bes t for Nursery dataset and other datasets having simil ar types of features.

D. Logistic regression

Logistic Regression is termed as a special case of Linear Regression which uses the concept of log odds to determine the relationship of the independent variables with the dependent variables using logit function. Thus, the outcome obtained is actually a probability. As per the thresholds defined, the data points can be classified into particular classes. It can be used for binary and multi class classification. It performs well when data isn't highly correlated to each other. Moreover, Logistic Regression requires the data to have a large observation set.

In the proposed model, Sonar dataset is used which was developed to determine if the sonar signals have bounced off metal cylinder or roughly cylinder rock. It consists of 60 independent variables and a dependent variable having two outcome classes (Mine, Rock). The independent variables are numerical and are scaled. Also, on plotting the heatmap of the correlation matrix of the variables it can be inferred from Fig. 11, that the data isn't highly correlated. Thus, there is no need of eliminating any of the features. If there were to be any highly correlated features, they could have been eliminated.

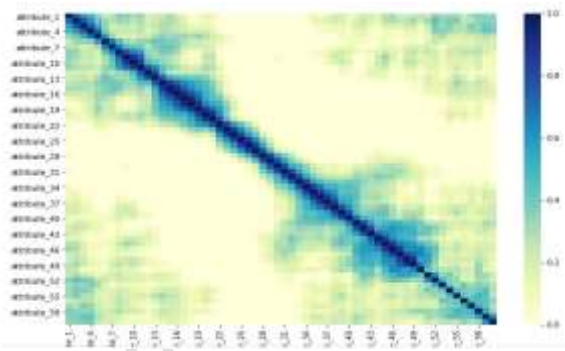


Fig.11. Heatmap of correlation of features

Furthermore, on observing the occurrence outcome classes in Fig. 12, the dataset can be deemed as balanced as the occurrence of classes is almost same. As the dataset is balanced, accuracy will be the correct performance measures. Also, as the dataset is balanced and the outcome is binary class even ROC curve can be considered to understand how good the classifier is.

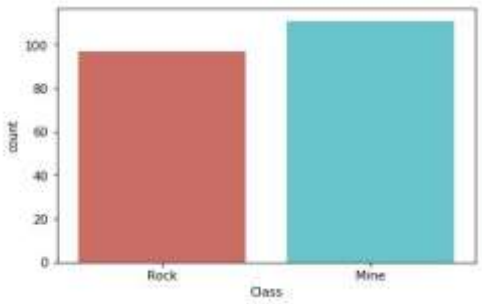


Fig. 12. Count of independent class variables

On comparing training and testing accuracy for all 60 features it was found that the model was overfitting. Thus, to determine the optimum number of features, recursive function elimination was applied. On plotting the training and testing accuracy for by using the method for recursive function elimination, the results can be seen in figure 13. It can be inferred that the model performs well at various depths but also overfits at majority of depths. Thus, considering overfitting and overall performance of the model, the maximum accuracy of 92.85 % is obtained at depth = 28.

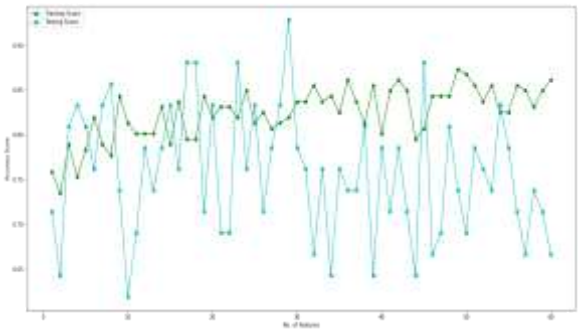


Fig. 13. Training v/s Testing accuracy using RFE

Subsequently, other models were constructed by app lying the other algorithms into consideration. All the models, even upon optimization, weren't able to gen erate results as good as the one given by Logistic Re gression. The performances of these models can be summarized as follows:

Table IV. Performance of models on Sonar Dataset

Datas et	Algorithm	Performance Measure: **Acc uracy**
Sonar Datas et	Naïve Bayes	66.66 %
	k- Nearest Neighbour s	87.45 %
	Decision Tree	83.33 %
	Logistic Regression	92.85 %
	Support Vector Mach ine	92.34 %

Thus, from the observations in table IV, it can be co ncluded that for the data like the one in Sonar Datas et, Logistic Regression will perform better than othe r basic classification models.

E. Support Vector Machine (SVM)

SVM is one of the most popular machine learning algorithms as it helps in producing significant accuracy. SVM does classification by finding optimal hyperplane that classifies data-points. SVM can work with high dimensions. It is also effective when numbers of samples are less than attributes. It can be applicable to most of machine learning tasks where slow training time is not a problem. Since it makes predictions based on only few points (support vectors), prediction time is quite fast in SVM.

Dataset that was chosen for this model was MNSIT Digit Recognizer dataset. Dataset contains 784 attributes (since 28*28-pixel image) with its corresponding labels. Problem was to build model that predicts digit based on the image. Since this dataset contained many attributes and also, spending more time for training was affordable as data may not be needed to be trained frequently, SVM was chosen for this problem. When data is not needed to be trained frequently and we have enough resources, usually SVM is the preferred algorithm especially when dimensions are high.

Dataset provided by MNSIT was already cleaned, so there was no need for pre-processing. Only thing that was needed to do was normalization of values. Main part for SVM algorithm is hyper-parameter tuning. There are many parameters which need to be taken into account such as kernel function, C, Gamma etc. First normal linear kernel with appropriate values of C and Gamma was tried since its execution time is faster. Then rbf kernel was (its performance is better than linear kernel but computation time is more) with appropriate values of C and Gamma was tried as training time was not our concern.

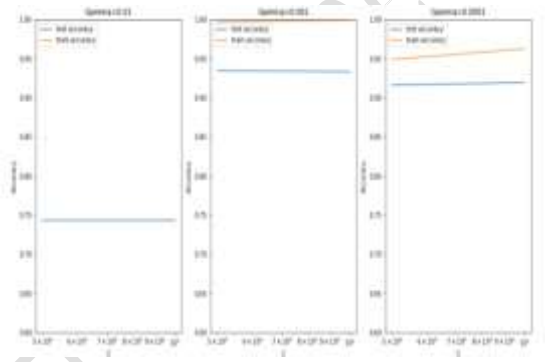


Fig. 14. Hyper parameter tuning

Since the dataset was well balanced (as it contained almost equal labels for all the digits) here accuracy would be better measure to judge performance of model rather than other measures. Accuracy in case of linear kernel was 91% while in case of rbf kernel was 94.4%.

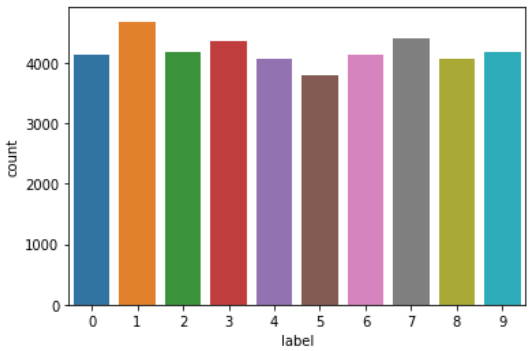


Fig. 15. Distribution of target variable (nearly balanced dataset

Subsequently, other models were constructed by app lying the other algorithms into consideration. All the models, even upon optimization, weren't able to gen erate results as good as the one given by SVM. The performances of these models can be summarized as follows:

Table V. Performance of models on MNSIT Digits Dataset

Dataset	Algorithm	Performance Measure: **Accuracy**
MNSIT Digits Dataset	Naïve Bayes	72.16 %
	k- Nearest Neighbours	91.17 %
	Decision Tree	80.95 %
	Logistic Regression	88.2 %
	Support Vector Machine	94.44 %

Thus, from the observations in table V, it can be concluded that for this dataset, our assumption that SVM will perform better than other basic classification models is verified.

IV. OBSERVATION

Table VI. Observation Table depicting performance measures

Classifier	Dataset	Performance Measures	Results
Naïve Bayes	Twitter Dataset	Precision	82.34%
SVM	MNSIT Digits dataset	Accuracy	94.4%
Decision Tree	Nursery Dataset	Accuracy	99.4%
KNN	Diabetes Dataset	Recall	76.56%
Logistic Regression	Sonar Dataset	Accuracy	92.85%

The table VI illustrates the performance of the basic classification algorithms on the respective datasets.

V. RESULT AND DISCUSSION

The approach explained in this paper considers time, quality and quantity of data available and importance of performance measures as the main criteria in carrying out the research. Thus, if the said approach is followed it would aid in time saving while there could be minor compromise in performance, in some cases. There could be other factors that could be considered while algorithm selection. Also, there could be cases wherein more than one algorithm could give similar results, in such cases, wither one of the basic algorithms could be chosen or ensemble method could be explored to derive better results from the chosen base classifiers.

Support Vector Machine (SVM) works comparatively well on all kinds of data in comparison with the other classification algorithms in consideration. The training time, in SVM, is greater than the testing time. Thus, it is more suitable in the cases where frequently training isn't required or if the training data isn't frequently updated. The training time required, in case of KNN, is less. But, it takes comparatively greater amount of time while making predictions. Thus, it shouldn't be preferred when the predictions are needed quickly. Naïve Bayes is an ideal choice for textual data. Logistic Regression performs best on highly correlated data. But, the only limitation, of Logistic regression, is that it requires large amount of data in dataset or else its performance is compromized. Decision Tree is preferred in case of categorical data. It could even work with numeric data, which can be converted into categories. But, the performance will vary according to the categorization of data. The training process is faster and the testing time depends on the depth of the tree. Decision Tree should not be preferred when new data arrives at a faster rate as even though the training time is comparatively faster, even a minuscule change in data can lead to construction of a completely new tree

VI.CONCLUSION

This paper provides an overview on how to choose between different classification algorithms, for a particular dataset, based on various factors like training time, prediction time, data distribution, number of features, expected outcome, type and quality of data, quantity/amount of data etc. that are important for a particular problem by implementing them on real datasets and synthetic datasets. It is interesting to note that approach for similar problem statements may differ depending on various constraints that are required to be fulfilled and thinking of developer. Sometimes, it is possible that more than one algorithm fits our requirements. In such cases, one can either construct multiple models using different base classifiers and then choosing the base classifier performing the best among other classifiers. Or, we can also use ensemble approach which considers single as well as multiple base classifiers.

Also, sometimes, it is possible that desired results might not be achieved by these standard methods. In such cases, various data pre-processing techniques can be applied which include dimensionality reduction, feature selection, sampling, etc. Moreover, this paper also elucidates the process of choosing the correct performance measure according to the use case or the business requirement. Not choosing the appropriate performance measures may lead to building of models which might not help to solve our problems; hence always the problem to be tackled needs to be considered while choosing the performance measure to evaluate the constructed model.

Journal of Engineering Sciences

VII. REFERENCES

[1] Lars Kotthoff, Ian P. Gent, Ian Miguel, "A Preliminary Evaluation of Machine Learning in Algorithm Selection for Search Problems", Proceedings, The Fourth International Symposium on Combinatorial Search (SoCS-2011)

[2] Hossin, Mohammad & M.N, Sulaiman, "A Review on Evaluation Metrics for Data Classification Evaluations". International Journal of Data Mining & Knowledge Management Process. 5. 01-11. 10.5121/ijdkp.2015.5201.

[3] Mariam Moustafa Reda, Dr Mohammad Nassef and Dr Akram Salah, "Factors Affecting Classification Algorithms Recommendation: A Survey", Computer Science Department, Faculty of Computers and Information, Cairo University, Giza, Egypt

[4] Cieslak D.A. and Chawla N.V., "Learning Decision Trees for Unbalanced Data". In: Daelemans W., Goethals B., Morik K. (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2008. Lecture Notes in Computer Science, vol 5211. Springer, Berlin, Heidelberg

[5] Vaishali Ganganwar, "An overview of classification algorithms for imbalanced datasets", International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 4, April 2012.

[6] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin "A practical guide to Support Vector Classification.", National Taiwan University, May 2016.

[7] Pouria Kaviani, Mrs. Sunita Dhotre, "A Short survey on Naïve Bayes Algorithm.", International Journal of Advance Engineering and Research Development, November 2017.

[8] R. Nancy Beaulah, "Performance Analysis of KNN on different types of attributes.", International Journal of Engineering and Computer Science, Volume 7, Issue 2, February 2018.

[9] Sayali Jadhav, H.P Channe, "Comparitive Study of K-NN, Naïve Bayes and Decision tree classification techniques.", International Journal of Science and research, Volume 5, Issue 1, January 2016.

[10] Bibhuprasad Sahu, Priyabrata Nayak, Archana panda, "Analysis of KNN with Naive Bayes, SVM and Naive Bayes Algorithms for Spam Mail Detection.",Ird India, Volume 5, Issue 4, 2016.

[11] Manuel Fernandez-Delgado, Eva Cernadas, Senen Barro, "Do we need Hundreds of Classifiers to solve Real Worlds Classification Problems.", Journal of Machine Learning Research, October 2014.

Journal of Engineering Sciences