Comparison of 5 classification algorithms: Decision Trees, Random Forest, Boosted Trees, Support Vector Machines and Neural Networks.

Ananthakrishnan V
Machine Learning Intern
AI Tech Systems
ai-techsystems.com
ananthakrishnanv94@gmail.com

***Abstract*— With the availability of huge amount of data and computational resources we are now living in a time which is being shaped by a change that is set forth by the constant growth and evolution of AI and Machine Learning. Machine Learning applications are highly automated and self-modifying which continue to improve over time with minimal human intervention as they learn with more data. Classifications problems are carried out frequently in Data Science domains and they have become relatively trivial because of Machine Learning. In it Supervised Machine learning algorithms are popular where a model is trained by mapping a function between an input and its output. In this paper we will compare five popular Supervised Machine Learning algorithms in terms of their working, speed of learning, accuracy, complexity, risk of overfitting and their specific domain of use.**

Keywords—Machine Learning(ML) , Deep Learning(DL), Classification, Neural Networks(NN), Support Vector Machine(SVM), Decision Tree(DT), Random Forest((RF), Gradient Boosted Decision Trees(GBDT), LSTMs

NOMENCLATURE

Neural Networks (NN), Decision Tree (DT), Random Forest (RF), Gradient Boosted Decision Tree (GBDT), Support Vector Machine (SVM), Human Activity Recognition (HAR).

I. INTRODUCTION

The field of Machine Learning has grown so much over the past few years and its progress can be largely credited to the availability of large amount of data and computational resources. Classification is an essential part of machine learning, pattern recognition and data analytics [1]. In machine learning, algorithms falls mainly under two categories, Supervised learning algorithms and Unsupervised learning algorithms. In supervised learning algorithm make predictions on given sets of samples. They search for patterns within the data labels assigned to the data points. Majority of practical machine learning uses supervised learning. Unsupervised learning algorithms learn the nature of data sets and try to find an underlying pattern or distribution in the data. This paper will talk about five supervised machine learning algorithms (DT, RF, GBDT, SVM and NN), their advantages, disadvantages, domain of use and finally compare their performance on Human Activity Recognition dataset. We will also analyse and compare complexity and speed of learning of each algorithm.

II. METHODOLOGY

The data used in this paper is Human Activity Recognition dataset. It was obtained from University of California Irwine's Machine Learning Repository. The dataset was chosen because of its ease of use and interpretability [1]. Also by using HAR data we could further analyse and understand the main differences between machine learning and deep learning. More about the dataset will be explained in detail in the next section. After obtaining the data, the next step involves preprocessing the data. This will include checking the data for NaN values and duplicates and also separating the data labels from the data. We also check for data imbalance. After the preprocessing part we try to visualise the dataset by using t-sne to get a better understanding of the dataset. Finally we

apply our algorithms on the dataset and compare each one with respect to the performance metric we chose which in this case is Accuray.

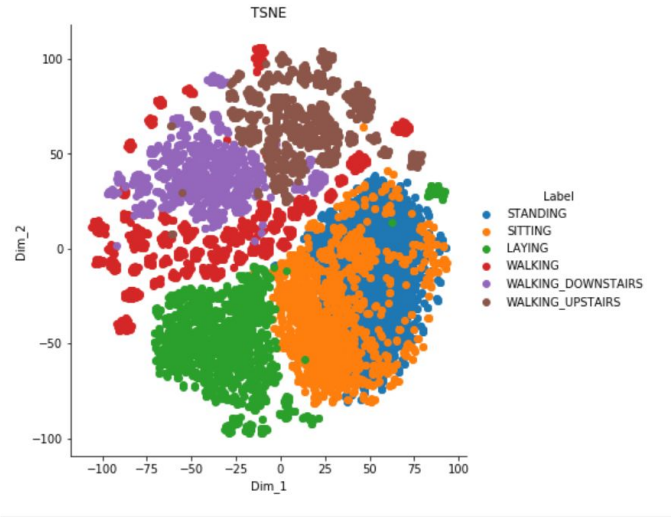

Fig.1. tsne plot of the data

III. DATASET

Human Activity Recognition dataset was made from data collected during an experiment in which 30 individuals were monitored. Each person performed six activities 1.Walking, 2.walking upstairs, 3.Walking downstairs, 4.Sitting, 5.Standing, 6.Laying wearing their smartphone. Using the accelerometer and gyroscope sensors from the smartphone 3-axial linear acceleration and 3-axial angular velocity was captured. For the convenience of our paper we have used different versions of the same datasets. Our original data is a raw time series sensor data of triaxial acceleration from accelerometer, estimated body acceleration and triaxial angular velocity from gyroscope. We will be using this raw data for deep learning [3].

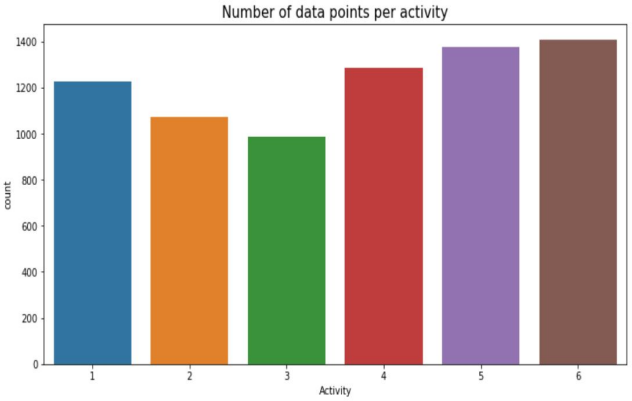

Fig.2. Number of datapoints per activity

For our classical machine learning models we will be using a modified version of the same data. The sensor signals were preprocessed by applying noise filters and the sampled in fixed width sliding windows(signal processing) of 2.56 seconds and 50% overlap. So our data will be having 128 readings per window [3]. On top of this signal processed data we have 561 domain expert engineered features as well. Some of the expert engineered features are :

- tBodyAcc-XYZ
- tGravityAcc-XYZ
- tBodyJerk-XYZ
- tBodyAccMag
- tGravityAccMag

IV. CLASSIFICATION TECHNIQUES

Decision Trees - Decision tree is a graphical representation that makes use of branching technique to show all the possible outcomes of a decision. They are highly interpretable and most similar to a nested if else classifier. They are a flowchart like structure and have a root node, internal nodes and leaf nodes. The internal nodes represent a condition or a test while the leaf nodes gives the outcome to the condition. Path from the root to a leaf represents classification rule [ ].

Advantages:

- Being a white box model, they are highly interpretable. Therefore they could be easily explained to anyone and are easy to comprehend.
- Since the split inside the tree happens based on a sample of data it is not prone to outliers. Decision trees are also not affected by missing values which will save us time during the preprocessing phase.

Drawbacks:

- They are high variance models ie, even a small change in the data can lead to a large change on the model. This puts decision trees at the risk of overfitting easily.

Application :

- They can be used in financial sectors for option pricing
- They are used in risk analysis in banking sectors to classify loan applicants by their payments.
- Rush University Medical Centre developed a tool called Guardian using Decision trees to find at risk patients and new disease trends [ ].

Random Forest - Random Forest is one of the most popular bagging technique used in machine learning. They use Decision Trees as base models, apply bagging (row sampling with replacement) on the DTs and do column sampling on the

data. In random forest we train a bunch of models with these random subset of data and the model will be a decision tree of reasonable depth. We find our result by taking a majority vote of all the base models. We want the DTs to be of high depth so as to make our base learners low bias, high variance models. This is because the row sampling, column sampling and aggregation will reduce the variance and we will end up with a low bias, low variance algorithm [ ].

Advantages :

- Since it uses a bunch of base models, it is not affected by outliers or overfitting and a robust model is the result.
- They do not require a large scale data preprocessing as they can handle numerical and categorical features without scaling or transformation.

Drawbacks :

- Cannot be used for many low latency real time applications as a large number of DTs tend to slow down the algorithm.

Application :

- Risk analysis in banks regarding loan applicants.
- Used to study and predict diseases and their on set on patients.
- Predicting pattern recognition in speech recognition, text and image data (large scale computer vision problems).

Gradient Boosted Decision Trees - In GBDT similar to RF we will be using a bunch of decision trees but instead of using a high variance, low bias model we will be using a high bias, low variance model. Instead of randomization and aggregation like in RF we will be using additive combining, ie, combine the model additively or sequentially to reduce the bias and obtain a low bias, low variance algorithm. In the case of GBDT we want out base models to be shallow so as to make them low variance models.

Advantages :

- Gradient boosting can be used in the field of Machine Learned Ranking (MLR).
- Each tree helps to reduce the error made by the previous tree which at the end results in a more expressive model.

Drawbacks :

- Training the model takes longer as each base learner is sequentially placed. Also its has 3 hyperparameters which further increase its training time.
- They are prone to overfitting, especially if the data is noisy.

Application :

- Works really well on unbalanced dataset such as DNA sequencing, credit card transactions and cyber security

Support Vector Machines - SVM is a supervised machine learning algorithm which is used for both classification and regression tasks. Most important point to note in case of SVM is that unlike most other classifiers it is not a probabilistic classifier but a deterministic one. In SVM, the decision boundary is not a line like in logistic regression but a plane called margin maximizing plane. Most common type of SVM is a linear SVM which as the name suggests can only do linear classification problems. SVMs can perform non-linear classification problems efficiently using a technique called kernel trick. These are functions that take low dimensional input space and transform them into higher dimensional space and thus converting non separable problems into separable problems by making use of some extremely complex data transformations [5].

Advantages :

- SVM offers best classification performance (accuracy) on training dataset.
- The kernel trick enables the user to built in expert knowledge about the problem via engineering the kernel.
- It does not overfit data.

Drawbacks :

- Kernel models can be quite sensitive to overfitting the model selection criterion.
- Hinger loss used in SVM causes sparsity

Application :

- It is commonly used in stock market forecasting, i.e, to compare the relative performance of stocks when compared to other socks.

Neural Networks - Artificial Neural Networks take inspiration from but not identical to our biological neural networks. Similar to neurons in our brain, ANNs have perceptrons. And just like a neuron has axons and dendrites a perceptron has inputs and output lines and each line will have weights associated with them. Just like in the case of a neuron the thicker the connection/weight the higher its importance. A single perceptron will take in inputs and give outputs. More complex model can be formed with more number and layers of perceptrons called Multi Layered Perceptrons(MLPs). Most popular and commonly used NNs are Convolutional Neural Networks (CNN), Recurrent Neural Networks(RNN), Long Short-Term Memory units (LSTM) etc. CNNs usually deal with image classification and LSTMs are used in case of time series data.

Advantages :

- Best in class at image classification and speech recognition.
- It gives the best performance metric when it comes to competing with ML classifiers.
- It can work with incomplete knowledge.

Drawbacks :

- Being a black box model its is not interpretable
- It relies heavily on computational hardware.
- They are hard to tune to give their optimal performance.

Applications :

- Image recognition and character detection.
- Forecasting in sales and financial domain.
- Cancer diagnosis in medical fields.

V. EXPERIMENTAL RESULTS

We can see that all 5 of our models have performed well on the dataset. Decision Tree classifier performed the lowest with a classification accuracy of 86.59 percent and it took the least time in training the model. From its confusion matrix we could see that for 'Lying' and 'Walking' the classifier did well as there are no miss classifications. The classifier's performance was average in ideally classifying the rest of the 4 class labels especially confusing between 'Sitting' and 'Standing. We used maximum tree depth as a hyper parameter

Our tree based ensemble models, Random Forest and Boosted Tree (GBDT) performed well on the dataset. RF gave an accuracy of 92.19 percent and GBDT gave an accuracy of 93.99 percent. With tree based ensemble models we achieved a 6 and a 7 percent increase in our performance metric compared to out DT. RF took less time to train but GBDT gave the better result, once again proving that a well tuned and trained boosted tree will perform better than a random forest.

We got the best accuracy on our support vector classifier. Both linear SVM and kernel SVMs were used and both the classifiers gave an accuracy of 96 percent.

On our Deep Learning model we got an accuracy metric of 89.95 percent which is close to 90 percent. We know that deep learning methods can automatically engineer features for us from raw time series data. LSTM models are known to easily overfit. Deep learning requires a large amount of data to build good models but considering the limited number of datapoints in our dataset (7352 data points) we can only train simple LSTM models. Our model is a single layer LSTM model with 32 parallel LSTM units each to itself. After the hidden layer we add a dropout layer with a dropout rate of 0.5. We add the dropout layer considering our number of parameters to be 5574 and number of datapoints to be 7352. Therefore dropout should be added to avoid overfitting. We used 'rmsprop' as our optimizer and categorical cross entropy as our loss as we are dealing with a multi class classification problem. Hence while dealing with a small amount of data, simplest DL architecture tends to do well.

TABLE. 1. Experimental Results

Classifiers	**Accuracy**
Decision Tree	86.59
Random Forest	92.19
GBDT	93.99
Linear SVM	96.50
Kernel SVM	96.26
LSTM	89.95

VI. CONCLUSION

From our experiment we could conclude that although all our models performed decently well, best accuracy was given by our support vector models. Tree based ensembles performed relatively well with our decision tree classifier giving the lowest performance. From the confusion matrix of each of our classifiers we could see that the majority of the misclassifications were made between sitting and standing. Hence by expert engineering some features that could further distinguish between sitting and standing, we could improve our models performance to a great deal.

Even though our DL model gave only a 90 percent accuracy it is safe to conclude that under the right circumstances we could get the best performance out of our deep learning models. Even with the simplest DL model with the right hyperparameter tuning of number of layers, number of neurons in each layer, the dropout rate, optimizer, the right distribution etc we could achieve a much higher accuracy than what we got with our classical machine learning models. With our handcrafted, expert engineered features the maximum accuracy our classical machine learning algorithm gave us was 96 percent. Without any prior knowledge of the domain, signal processing or feature engineering, just a simple LSTM model gave us an accuracy of 90 percent. Therefore it is safe to assume that with a good amount of data, with just the raw time series dataset without any feature engineering we could achieve the same or even better accuracy than our machine learning models with expert engineered features.

REFERENCES

[1] H. I. Bulbul and Ö. Unsal, "Comparison of Classification Techniques used in Machine Learning as Applied on Vocational Guidance Data," *2011 10th International Conference on Machine Learning and Applications and Workshops*, Honolulu, HI, 2011, pp. 298-301.

doi: 10.1109/ICMLA.2011.49

[2] A. Singh, N. Thakur and A. Sharma, "A review of supervised machine learning algorithms," *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, 2016, pp. 1310-1315

[3] K. Kokkinidis, T. Mastoras, A. Tsagaris and P. Fotaris, "An empirical comparison of machine learning techniques for chant classification," *2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, Thessaloniki, 2018, pp. 1-4.

doi: 10.1109/MOCAST.2018.8376596

[4] archive.ics.ucii.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

[5] analyticsvidya.com

[6] Aplliedaicourse.com