Enhancement of Urban Sound Classification Using Various Feature Extraction Techniques

Afshankaleem*, I. Santi Prabha

Abstract--- In this paper we describe few methods of extracting features from sound data, one commonly used feature extraction technique in speech recognition is isolating the Mel Frequencies Cepstral Coefficients (MFCC). The accuracy of speech recognition systems, to a large extent, depends on the feature sets used for representing the recorded speech data. It has been a continuous process to derive better feature sets for more accurate speech recognition using ASR (Automatic Speech Recognition) systems. Many feature sets and their different combinations have been tried to achieve better accuracy but a feature set providing completely accurate results has not yet been formulated. These large feature sets consume significant amount of memory, together with computing and power requirements and they do not always contribute to improve the recognition rate. There are few commonly used features extraction methods, such as Mel-scaled spectrogram, Chroma gram, spectral-contrast, and the tonal centroid features We go on to detail the effectiveness of different models on each method, including tests of Random Forests, Naïve Bayes,J48, SVM, Machines architectures

Keywords--- Mel Banks Cepstral Coefficients (MFCC), Sound Classification, Feature Extraction.

I. INTRODUCTION

URBAN sound classification has been a field of growing research. [1] In particular, The Urban Sound Planner brings a different perspective and contributions to the process of delivering well-designed cities that work for people rather than harm them. This is achieved by improving the quality of sonic urban environments, not simply trying to make things quieter, but proactively designing to avoid noise generation and defining policies and strategies to value, introduce and preserve the characteristics of a good sonic environment. The sonic analysis on environment sounds has generated increasing researches because of its multiple applications to large-scale content-based multimedia indexing and retrieval. Not only are there scarce works on environment sound, but also very few database for labelled environment audio data. One of the few free large sound data is the UrbanSound 8k dataset created by Justin Salamon, Christopher Jacoby and Juan Pablo Bello in 2014. The Urban 8k Sound dataset is unique for its classification is not just based on the auditory scene type such as nature, human, animal, but on the sources of sound, such as dog bark, car horn.

II. DATASET

We need a labelled dataset that we can feed into machine learning algorithm .This dataset contains 8732 labelled sound excerpts; each sample has a duration of 4 seconds of urban sounds from 10 classes: air Conditioner, carhorn, children playing, dog bark, drilling, engingeidling, gunshot, jackhammer, siren, and street music. The classes are drawn from the urban sound taxonomy.

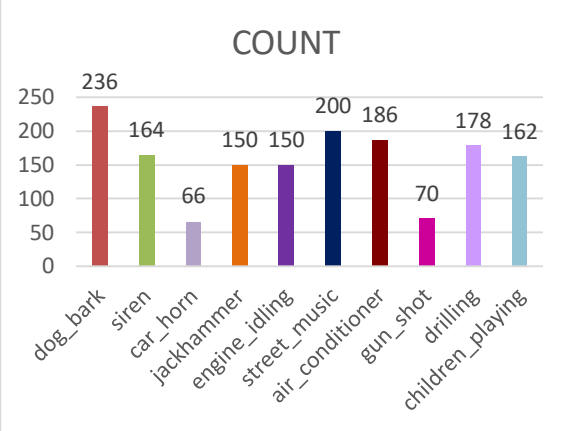


Figure 1: Shows Different classes of data set

The files are pre-sorted into ten folds (folders named fold1-fold10) to help in the reproduction of and comparison with the automatic classification results.

We all got exposed to different sounds every day. Like, the sound of car horns, siren and music etc. How about teaching computer to classify such sounds automatically into categories!. we will learn techniques to classify urban sounds into categories using machine learning. when it comes to sound, feature extraction is not quite straightforward. First see what features can be extracted from sound data and how easy it is to extract such features in Python using open source library called Librosa.

Likewise, Librosa provide handy method for wave and MFCC spectrogram plotting. By looking at the plots shown in Figure 1, 2 and 3, we can see apparent differences between sound clips of different classes. [1]

III. FEATURE EXTRACTION

To extract the useful features from sound data, we will use *Librosa* library. It provides several methods to extract different features from the sound clips. The entire process of extracting MFC C features is illustrated in Fig. 2. In this research, we utilize feature set consists of 12 melcepstrum coefficients, one logenergy coefficient, 13 delta coefficients an d 13 delta2 coefficient per frame which in total 3 9 coefficients. We are going to use below mentioned methods to extract various features:

Manuscript received February 01, 2019

Afshankaleem*, Sr.Assistant Professor, Dept of ECE, MJCET, Hyderabad, Telangana, India. (e-mail: afshankaleem@gmail.com)

Dr.I. Santi Prabha, Professor, Dept. of ECE, JNTUCE, JNTUK, Kakinada, Andhra Pradesh, India. (e-mail: santiprabha@yahoo.com)

- ***Mel-spectrogram*:** Compute a Mel-scaled power spectrogram
- ***Mfcc*:** Mel-frequency cepstral coefficients
- ***Chorma-stft*:** Compute a chromagram from a waveform or Cpower spectrogram
- ***Spectral contrast*:** Compute spectral contrast, using method defined in [6]
- ***Tonnetz*:** Computes the tonal centroid features (tonnetz), following the method of [7]

To make the process of feature extraction from sound clips easy, two helper methods are defined. First parseaudio files which takes parent directory name, subdirectories within parent directory and file extension (default is .wav) as input. It then iterates over all the files within subdirectories and call second helper function extract feature. It takes file path as input, read the file by calling *librosa.load* method, extract and return features discussed above. These two methods are all that is required to convert raw sound clips into informative features (along with a class label for each sound clip) that we can directly feed into our classifier. Remember, the class label of each sound clip is in the file name. For example, if the file name is *108041-9-0-4.wav* then the class label will be 9. Doing string split by – and taking the second item of the array will give us the class label.

One commonly used feature extraction technique in speech recognition is isolating the Mel Frequencies Cepstral Coefficients (MFCC). It is also widely used in environmental sound analysis and has become competitive baseline for bench marking new techniques. Steps to extract

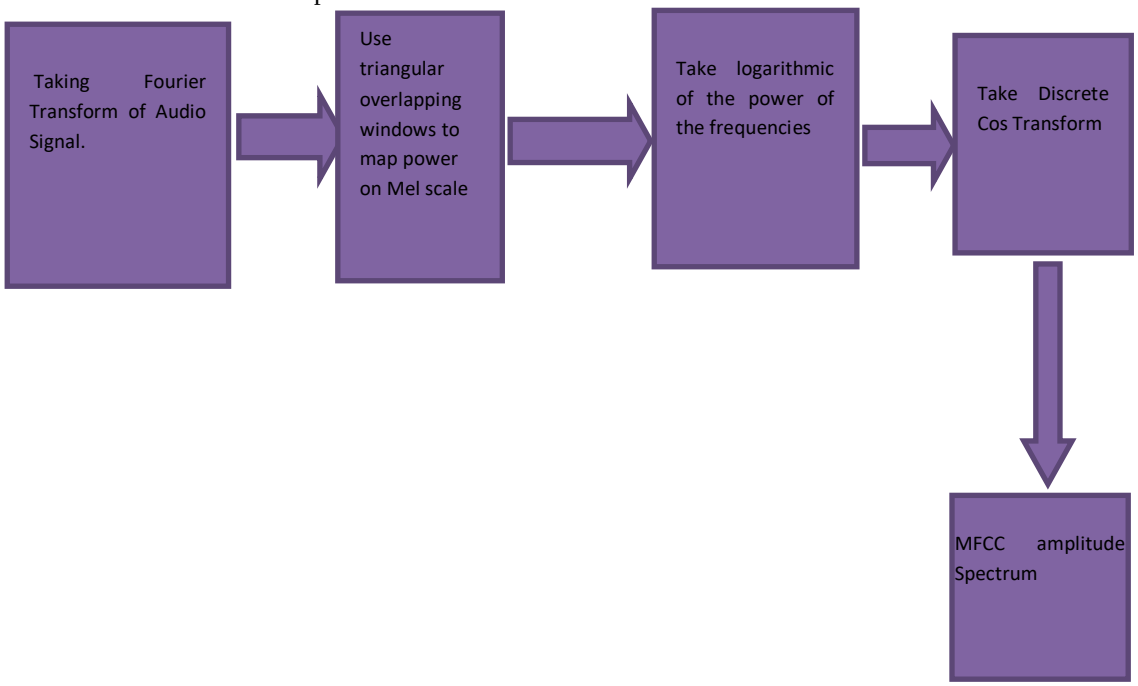
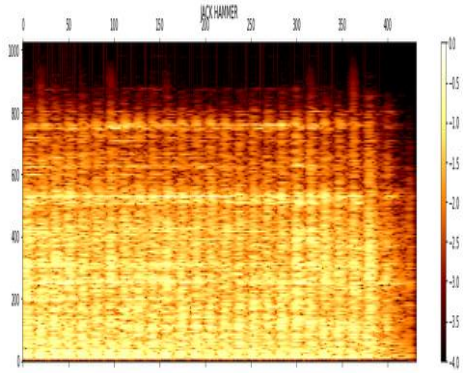


Figure 2: MFCC Block Diagram

1. Frame the signal into short frames of 20 – 40 milliseconds. Each frame overlaps 50% of its neighbour frames.
2. For each frame calculate the periodogram estimate of the power spectrum. This step records what frequencies are presented in each frame.



3. "Apply the Mel filterbank to the power spectra, sum the energy in each filter. This step takes clumps of periodogram bins and sum them up to further reduce the number of features.
4. "Take the logarithm of all filter bank energies. This step is motivated by the fact that human does not perceive loudness in a linear scale.
5. "Take the discrete cosine transformation (DCT) of the log filterbank energies. This step de-correlates overlapping frames.
6. "Keep DCT coefficients 2-13, discard the rest. Experiment results show that dropping the coefficient above the 13th improve performance in Sound rec

There are other commonly used feature extraction methods, such as Mel-scaled spectrogram, Chromagram, spectral-contrast, and the tonal centroid features.

Mel Spectrogram: A spectrogram is the visual representation of the spectrum of frequencies of sound or other signal as they vary with time. These are also known as voicegrams, sonographs or voiceprints.

It is a visual method of depicting the strength of the signal or loudness of a signal over time at various frequencies that are present in a particular waveform. When we use a nonlinear mel scale of frequency, we obtain the Mel Spectrogram. A mel scale is a scale of pitches judged by listeners to be equal in distance from one another

Chroma STFT: The term chromagram or chroma feature closely relates to the 12 different classes of pitch. These are also referred to as pitch class profiles and a powerful tool for the analysis of music whose pitches can be meaningfully categorized. One main property of chromagramis that they capture themelogi candharmonic characteristics music 14 while being robust and agile to changes in instrumentation and timbre. In this we usually calculate a chromagram from the waveform of the power spectrum.

Spectral Contrast: Octave based Spectral Contrast consider the spectral peaks, spectral valleys and their difference in each sub-band. In simple terms, it roughly represents the relative distribution of the harmonic and non-harmonic components in the spectrum. Other features like MFCC , take the average of the spectral distribution in each subband and are thus prone to lose valuable spectral information.

Tonnetz: This is also feature which detects the changes in the harmonic content of the musical audio signals. A peak in the detection function represents that a transition was made from one harmonically stable region to another. It has been observed that the algorithm can successfully detect harmonic changes such as chord boundaries in the polyphonic audio recordings

The Librosa library has functions to extract all the above characteristics. We ended up extracting a total of 193 characteristics (features) for each sample using these methods. The second category of our feature extraction approaches is filter banks. This approach allows us to keep the time-series attribute of the raw data. (e.g. The extracted series of features are also time series) Computing filter banks and MFCCs involve somewhat the same procedure, where in both cases filter banks are computed and with a few more extra steps MFCCs can be obtained.)[4] To obtain MFCCs, a Discrete Cosine Transform (DCT) is applied to the filter banks retaining a number of the resulting coefficients. Because of the way we partition each sample into overlapping frames, each filter banks we extract is highly correlated with its neighbour filterbanks. This autocorrelation can be problematic for some classifiers such as the hidden Markov Chain It might be beneficial for the classifiers to learn directly from the signal in the time domain.

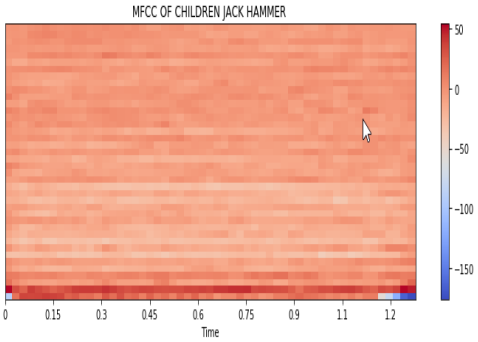


Figure 3: Mfcc of Jack Hammer

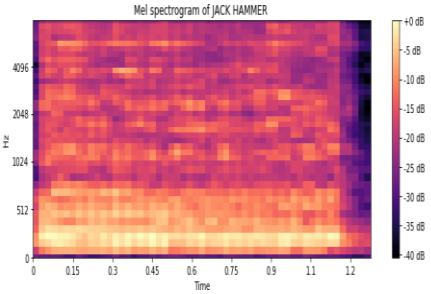


Figure 4: MelSpectrogrm of Jack Hammer

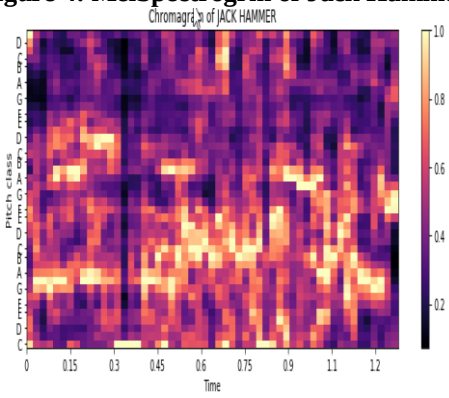


Figure 5: Chromogram of Jack Hammer

The challenge we faced in extracting the filterbanks features are different-sized raw data. Since our samples include sound excerpts of different length, resolution, and number of channels. The shape of filterbank features also varies across samples. Because neither Sklearn nor Tensor flow allows varied data shape, we would need to make the size of extracted features identical across samples. Our first solution is zero padding, making the shorter signals as long as the longest ones. Our second solution is to cut each sample into the same number of windows (frames), so each sample will have the same number of windows regardless of the length of the original sound. Shorter samples will have a frame partition that each frame overlaps with others more.

It's important to consider dimensionality when doing machine learning on audio data. Even though Librosa converts recordings to mono and down-samples to 22050 hz, (considered a good compromise quality for audio analysis), you'll still end up with 22050 data values for every second of your recording.

Dimensionality reduction, an insight that will extract just the salient information from audio samples. This can be done by concatenating the results of the 5 extractions listed above to give a consistent feature vector of 193 values for every audio clip processed

IV. MODELS

Urban Sound Classification (USC)

The general workflow of an USC (urban sound classification) system is usually divided into two major steps in figure --.

1) Feature Extraction
2) Classification.

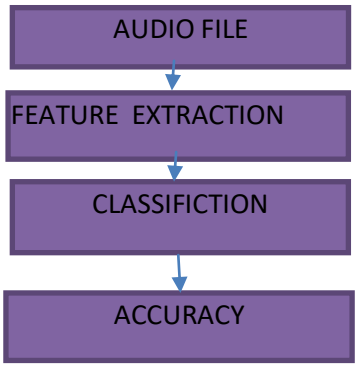


Figure 6: Work flow of USC

Machine Learning Classification Algorithms

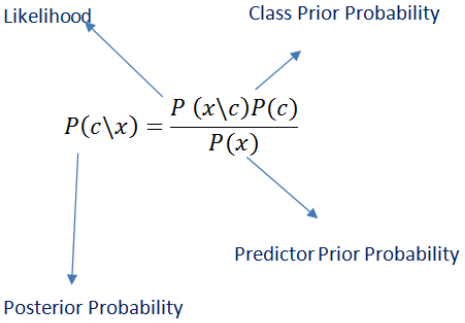
Classification is a supervised data mining technique that assigns labels to a collection of data in order to get more accurate predictions and analysis. The USC is the task to assigns label to audio data to know the label of audio by using trained classifier. The labels for unknown audio data may different according to the application domain. In this proposed work, Naïve Bayees. Random Forest Analysis are used to make the analysis of very large datasets effective.

Naive Bayes‘, which can be extremely fast relative to other classification algorithms. It works on Bayes theorem of probability to predict the class of unknown data set.

It is a classification technique based on Bayes’ Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as ‘Naive’.

Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:



$$P(c\backslash X) = P(x_1\backslash c) \times P(x_2\backslash c) \times \ldots\ldots\ldots\ldots \times P(x_n\backslash c) \times P(C)\ldots\ldots$$

Above,

- $P(c|x)$ is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
- $P(c)$ is the prior probability of *class*.
- $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*.
- $P(x)$ is the prior probability of *predictor*.

Pros:

It is easy and fast to predict class of test data set. It also performs well in multi class prediction

When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.It performs well in case of categorical input variables compared to numerical variable(s). For numerical variable, normal distribution is assumed (bell curve, which is a strong assumption).

Cons:

If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as “Zero Frequency”. To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.

On the other side naive Bayes is also known as a bad estimator, so the probability outputs from predict probably are not to be taken too seriously.

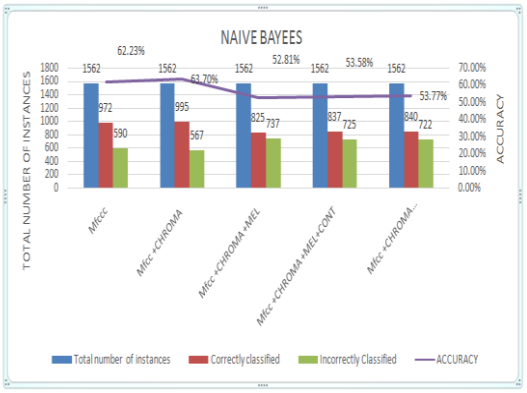
Another limitation of Naive Bayes is the assumption of independent predictors. In real life, it is almost impossible that we get a set of predictors which are completely independent.

V. RESULTS

Table 1(a): Naïve Bayes

CLASIFFIERS	Summary	Mfccc		Mfcc +CHROMA		Mfcc +CHROMA +MEL		Mfcc +CHROMA +MEL+CONT		Mfcc +CHROMA +MEL+CONT+TONNETS	
NAÏVE BAYEES	Total number of instances	1562		1562		1562		1562		1562	
	Correctly classified	972	62.23%	995	63.70%	825	52.81%	837	53.58%	840	53.77%
	Incorrectly Classified	590	37.77%	567	36.30%	737	47.18%	725	46.41%	722	46.22%
	Statistics										
	Time to build model	0.02sec		0.01sec		0.03sec		0.02sec		0.03sec	
	Mean Absolute Error	0.078		0.0725		0.0942		0.092		0.0925	
	Root Mean Square Error	0.2505		0.25		0.3065		0.3037		0.303	

Table: 1(b): Naïve Bayees Analysis



J48 (DECISION TREE)

Decision Tree Algorithm is to find out the way the attributes-vector behaves for a number of instances. Also, on

the bases of the training instances the classes for the newly generated instances are being found This algorithm generates the rules for the prediction of the target variable.

With the help of tree classification algorithm, the critical distribution of the data is easily understandable. J48is an extension of ID3. The additional features of J48 are accounting for missing values, decision trees pruning, continuous attribute value ranges, derivation of rules, etc.

The WEKA tool provides a number of options associated with tree pruning. In case of potential over fitting pruning can be used as a tool for précising. In other algorithms the classification is performed recursively till every single leaf is pure, that is the classification of the data should be as perfect as possible. This algorithm it generates the rules from which particular identity of that data is generated. The objective is progressively generalization of a decision tree until it gains equilibrium of flexibility and accuracy.

Steps in the Algorithm:

(i) In case the instances belong to the same class the tree represents a leaf so the leaf is returned by labelling with the same class.

(ii) In case the instances belong to the same class the tree represents a leaf so the leaf is returned by labelling with the same class.

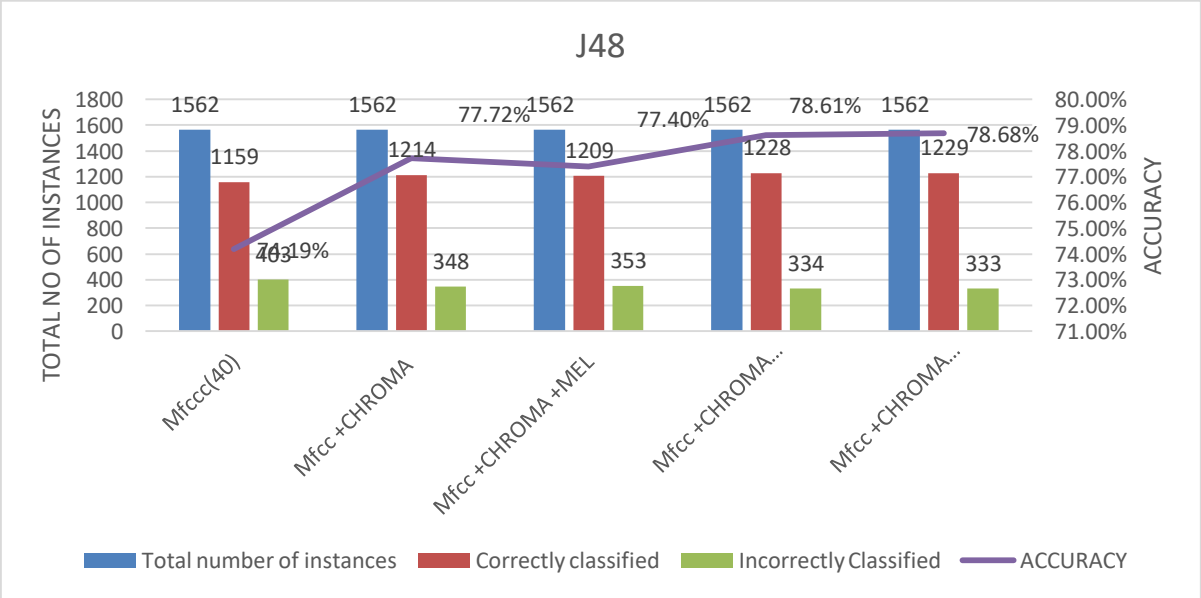
(iii) The potential information is calculated for every attribute, given by a test on the attribute. Then the gain in information is calculated that would result from a test on the attribute.

(iv) Then the best attribute is found on the basis of the present selection criterion and that attribute selected for branching.

Table 2(a):-J48 Analysis

CLASIFFIERS	Summary	Mfccc		Mfcc +CHROMA		Mfcc +CHROMA +MEL		Mfcc +CHROMA +MEL+CONT		Mfcc +CHROMA +MEL+CONT+TONNETS	
J48	Total number of instances	1562		1562		1562		1562		1562	
	Correctly classified	1159	74.19%	1214	77.72%	1209	77.40%	1228	78.61%	1229	78.68%
	Incorrectly Classified	403	25.41%	348	22.27%	353	22.59%	334	21.38%	333	21.31%
	Statistics										
	Time to build model	0.38sec		0.14sec		0.48sec		0.55sec		0.59sec	
	Mean Absolute Error	0.054		0.04		0.04		0.04		0.045	
	Root Mean Square Error	0.021		0.2		0.2		0.199		0.199	

Table 2(b): J48 Analysis



Support Vector Machines (SMO)

Support Vector Machines were developed for binary classification problems, although extensions to the technique have been made to support multi-class classification and regression problems. The algorithm is often referred to as SVM for short.SVM was developed for numerical input variables, although will automatically convert nominal values to numerical values. Input data is also normalized before being used.

SVM work by finding a line that best separates the data into the two groups.

This is done using an optimization process that only considers those data instances in the training dataset that are closest to the line that best separates the classes. The instances are called support vectors, hence the name of the technique.

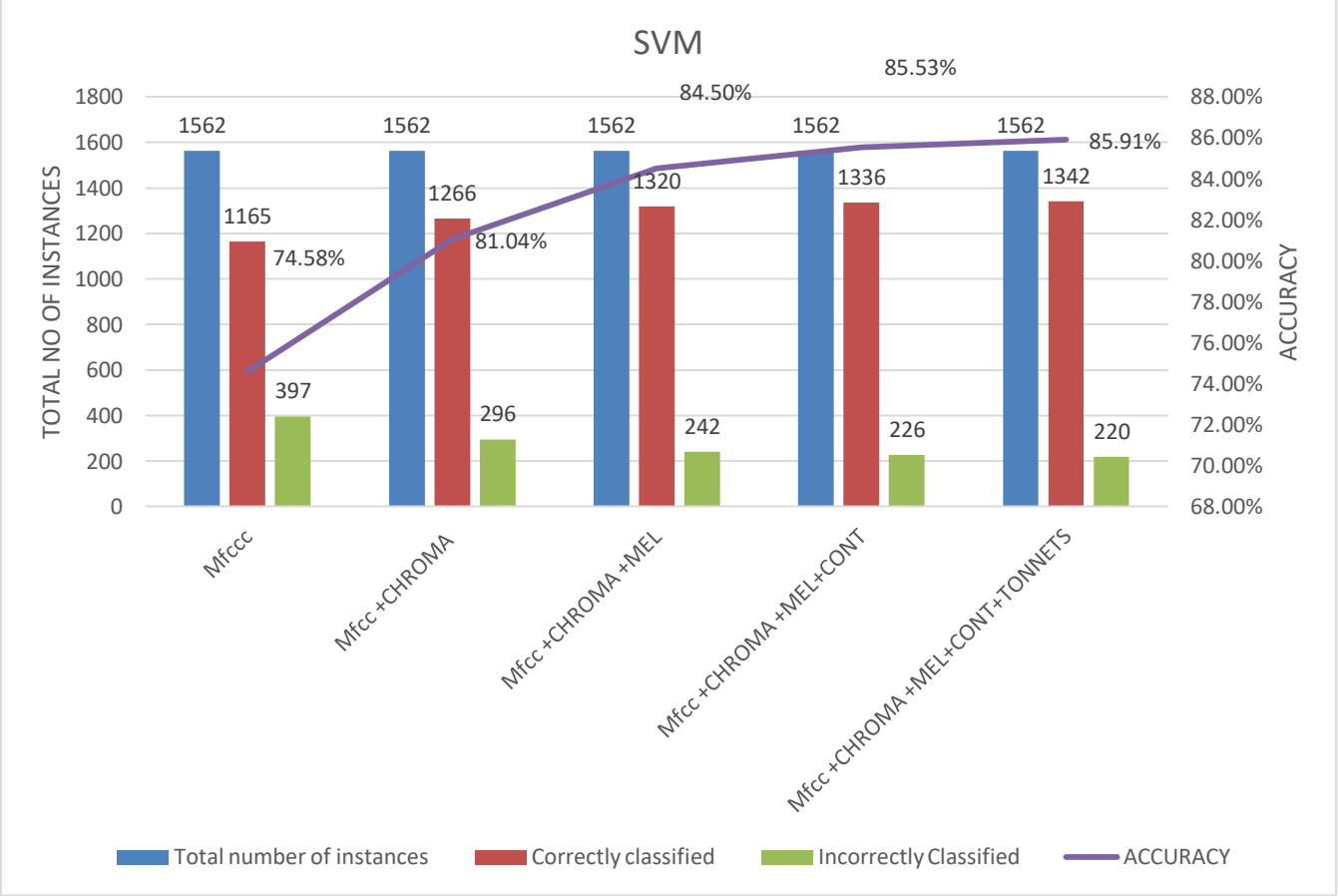
Few datasets can be separated with just a straight line. Sometimes a line with curves or even polygonal regions need to be marked out. This is achieved with SVM by projecting the data into a higher dimensional space in order to draw the lines and make predictions. Different kernels can be used to control the projection and the amount of flexibility in separating the classes.

SMO refers to the specific efficient optimization algorithm used inside the SVM implementation, which stands for Sequential Minimal Optimization.

Table: 3(a): SVM

CLASIFFIERS	Summary	Mfccc		Mfcc +CHROMA		Mfcc +CHROMA +MEL		Mfcc +CHROMA +MEL+CONT		Mfcc +CHROMA +MEL+CONT+TONNETS	
SVM	Total number of instances	1562		1562		1562		1562		1562	
	Correctly classified	1165	74.58%	1266	81.04%	1320	84.50%	1336	85.53%	1342	85.91%
	Incorrectly Classified	397	25.80%	296	18.95%	242	15.49%	226	14.46%	220	14.08%
	Statistics										
	Time to build model	0.53 sec		0.23sec		0.38sec		0.38sec		0.56sec	
	Mean Absolute Error	0.054		0.16		0.16		0.16		0.16	
	Root Mean Square Error	0.021		0.27		0.27		0.27		0.27	

Table 3(b): SVM ANAYSIS



Random Forest: is a supervised learning algorithm. Like you can already see from it's name, it creates a forest and makes it somehow random. The forestit builds, is an ensemble of Decision Trees, most of the time trained with the "bagging" method. The general idea of the bagging method is that a combination of learning models increases the overall result

To say it in simple words: Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. One big advantage of random forest is, that it can be used for both classification and regression problems, which form the majority of current machine learning systems.

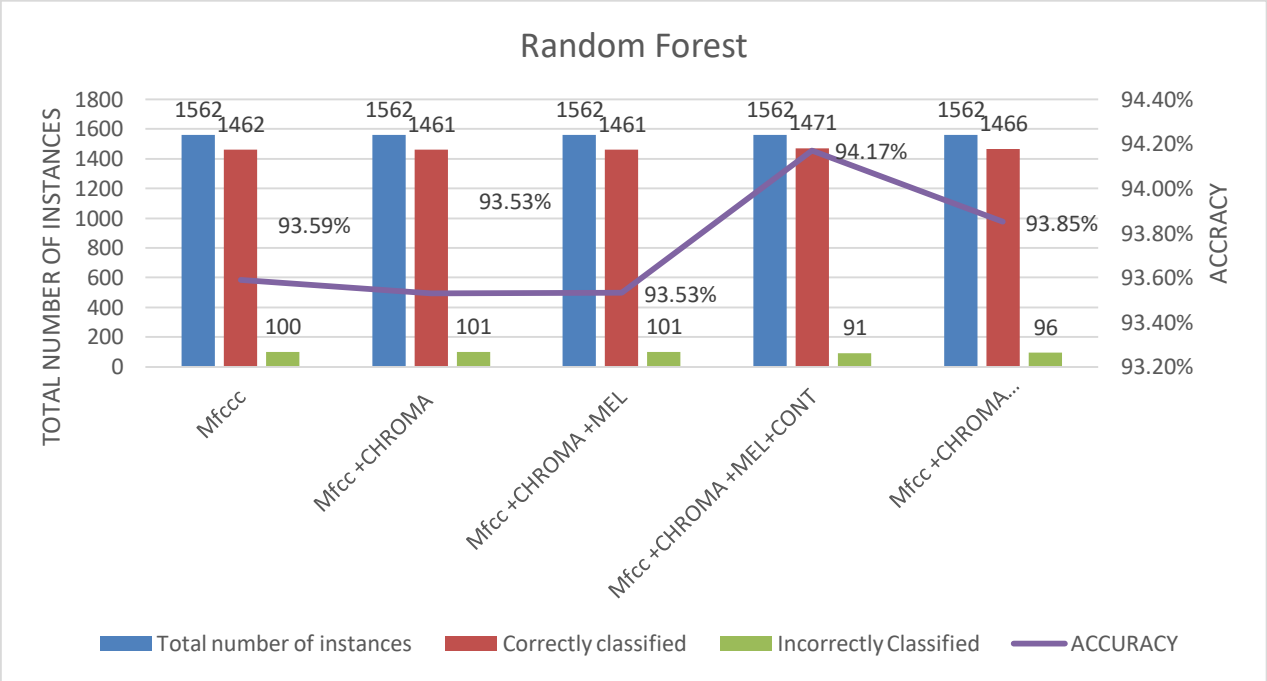
Random Forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model.

Therefore, in Random Forest, only a random subset of the features is taken into consideration by the algorithm for splitting a node. You can even make trees more random, by additionally using random thresholds for each feature rather than searching for the best possible thresholds (like a normal decision.

Table 4(a): Random Forest

CLASIFFIERS	Summary	Mfccc(40)		Mfcc +CHROMA		Mfcc +CHROMA +MEL		Mfcc +CHROMA +MEL+CONT		Mfcc +CHROMA +MEL+CONT+TONNETS	
RANDOM FOREST	Total number of instances	1562		1562		1562		1562		1562	
	Correctly classified	1462	93.59%	1461	93.53%	1461	93.53%	1471	94.17%	1466	93.85%
	Incorrectly Classified	100	6.04%	101	6.04%	101	6.46%	91	5.82%	96	6.15%
	Statistics										
	Time to build model	0.86sec		0.86sec		1.25 sec		1.2sec		1.2sec	
	Mean Absolute Error	0.068		0.068		0.622		0.061		0.623	
	Root Mean Square Error	0.147		0.146		0.139		0.137		0.138	

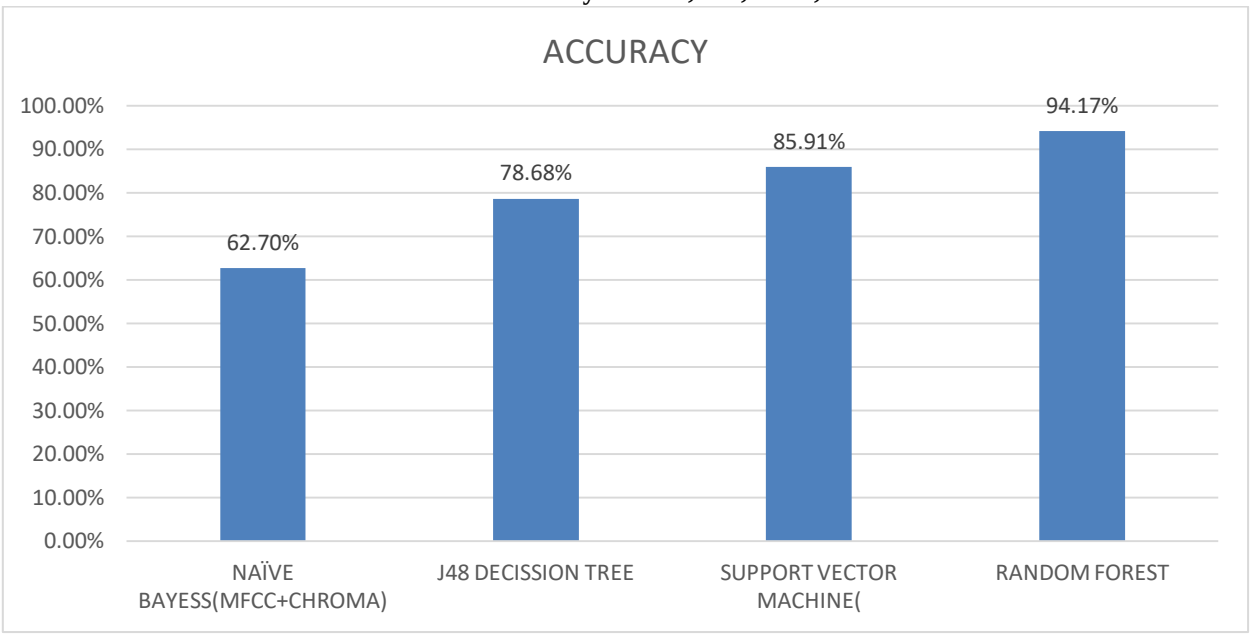
Table 4(b): Random Forest



VI. FINAL RESULTS

Comparison of average classification accuracy for MFCC features with different classifiers over10 fold cross validation.

Table 5: Accuracy of NAB, J48, SVM, RF



VII. CONCLUSION

MFCC features is one of the good features which have acceptable average classification accuracy with different machine learning classifications. Our comparative study of different classifiers with MFCC feature has reasonable classification accuracy to solve the difficulties in choosing best classifiers for USC. The moment statistic of MFCC features called MFCC-moment also has acceptable classification accuracy for USC. For future work, different features will be investigated for a set of innovative features for best classification accuracy in USC. Although both MFCC and MFCC-moment features have acceptable classification accuracies with machine learning classifiers, we need to try to improve the classification accuracy of USC by considering other signal processing techniques and classifier.

ACKNOWLEDGMENTS

Our thanks to Justin Salamon, Christopher Jacoby, and Juan Pablo Bello for creating the UrbanSound8K dataset.

REFERENCES

[1] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. 2014. A Dataset and Taxonomy for Urban Sound Research. In Proceedings of the 22nd ACMinternational conference on Multimedia (MM '14). ACM, New York, NY,

[2] James Lyons. 2013. Mel Frequency Cepstral Coefficient (MFCC) tutorial. In Practical Cryptography Online. URL=http://www.practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficientsmfccs/

[3] Hibare, Rekha, and Anup Vibhute. "Feature Extraction Techniques in Speech Processing: A Survey." International Journal of Computer Applications 107.5 (2014)

[4] Dan-Ning hang*, Lie Lu**, Hong-Jiang Zhang**, Jian-Hua Tao*, Lian-Hong Cui* "Music type Classification by Spectral contrast feature" Proceedings. IEEE International Conference on Multimedia and Expo

[5] Aaqid Sayeed. 2016 Urban Sound Classification, Part I, Part II.

[6] Hibare, Rekha, and Anup Vibhute. "Feature Extraction Techniques in Speech Processing: A Survey." International Journal of Computer Applications 107.5 (2014).

[7] Dan-Ning hang*, Lie Lu**, Hong-Jiang Zhang**, Jian-Hua Tao*, Lian-Hong Cui*"Music type Classification by Spectral contrastfeature" Proceedings. IEEE International Conference on Multimediaand Expo

[8] Christopher Harte and Mark Sandler Martin Gasser "Detecting Harmonic Change In Musical Audio "AMCMM'06,October 27, 2006, Santa Barbara, California, USA.

[9] L. Breiman, "Pasting small votes for classification in large databases and on-line", Machine Learning, 36(1), 85-103, 1999.

[10] Sonia Suuny, David Peter S , K. Poulose Jacob, "Performance of Different Classifiers In Speech Recognition", 2013 IJRET.

[11] Justin Salmon, Christopher Jacoby, Juan Pablo Bello"Music and Audio research Laboratory, New York University, Centre of Urban Science and Progress",2014, USA

[12] Huan Zhou, Ying Song , Haiyan Shu,"Smart Energy and Environment Cluster Institute for Infocomm Research", IEEE Region Conference(TENCON), Malaysia,November 5-8,2017

[13] N. R. Fatahillah, P.Suryati and C. Haryawan, "Implementation of Naive Bayes classifier algorithm on social media (Twitter) to the teaching of Indonesian hate speech," 2017 International Conference on Sustainable Information

[14] N. Albarakati and V. Kecman, "Fast neural network algorithm for solving classification tasks: Batch error back-propagation algorithm," 2013 Proceedings of IEEE Southeastcon, Jacksonville, FL, 2013, pp. 1-8

[15] A. B. Kandali, A. Routray and T. K. Basu, "Emotion recognition from Assamese speeches using features and GMM classifier," TENCON 2008-2008 IEEE Region 10 Conference, Hyderabad, 2008, pp. 1-5.