Int. J. Adv. Eng. Pure Sci. 2021, ASYU 2020 Special Issue: 17-27
DOI: 10.7240/jeps.896427

RESEARCH ARTICLE / ARAŞTIRMA MAKALESİ

Speaker Accent Recognition Using MFCC Feature Extraction and Machine Learning Algorithms

MFCC Öznitelik Çıkarım Tekniği ve Makine Öğrenmesi Algoritmaları Kullanılarak Konuşmacı Aksanı Tanıma

Ahmet Aytuğ AYRANCI [1,2], **Sergen ATAY**[2], **Tülay YILDIRIM**[2]

[1] *İstanbul Kültür University, Electrical and Electronics Engineering, 34722, İstanbul, Türkiye*
[2] *Yıldız Technical University, Electronics and Communication Engineering, 34722, İstanbul, Türkiye*

Abstract
Speech and speaker recognition systems aim to analyze parametric information contained in the human voice and recognize it at the highest possible rate. One of the most important features in the voice for successful speaker recognition is the speaker's accent. Speaker accent recognition systems are based on the analysis of patterns such as the way that the speaker speaks and the word choice he uses while speaking. In this study, the data obtained by the MFCC feature extraction technique from voice signals of 367 speakers with total 7 different accents were used. The data of 330 speakers in the data set were taken from the "Speaker Accent Recognition" dataset in the UC Irvine Machine Learning (ML) open data repository. The data of the other 37 speakers were obtained by converting the voice recordings in the "Speaker Accent Archive" dataset created by George Mason University into data using the MFCC feature extraction technique. 9 ML classification algorithms were used for the designed speaker accent recognition system. Also, the k-fold cross-validation technique was used to test the data set independently. In this way, the performance of ML algorithms is shown when the dataset is divided into a k number of parts. Information about the classification algorithms used in the designed system and the hyperparameter optimizations made in these algorithms are also given. The performances of the classification algorithms are shown with using performance metrics.
Keywords: Mel-frequency cepstral coefficients, machine learning, speaker accent recognition, feature extraction

Öz
Konuşma ve konuşmacı tanıma sistemlerinde insan sesinin içerdiği parametrik bilginin sistem tarafından analiz edilip en yüksek başarı oranında tanınması hedeflenmektedir. Konuşmacı tanımanın başarılı bir şekilde yapılabilmesi için ses içerisindeki en önemli özelliklerden bir tanesi konuşmacının aksanıdır. Konuşmacı aksanı tanıma sistemleri konuşan kişinin konuşma şekli ve konuşurken kullandığı kelime seçimi gibi örüntülerin analiz edilerek tanınmasına dayanmaktadır. Konuşmacının ses sinyalinden gerekli öznitelik bilgilerini elde etmek için Mel-Frekans Kepstral Katsayıları (MFCC) öznitelik çıkarım tekniği kullanılmıştır. Bu çalışmada 12 katsayılı MFCC tekniği ile toplamda 7 farklı aksana sahip 367 konuşmacıya ait ses sinyallerinden elde edilen veriler kullanılmıştır. Kullanılan veri setindeki 330 konuşmacıya ait veriler UC Irvine Makine Öğrenmesi (ML) açık veri kaynağındaki “Speaker Accent Recognition” veri setinden alınmıştır. Diğer 37 konuşmacının verisi ise George Mason Üniversitesi tarafından oluşturulan “Speaker Accent Archive” veri setindeki ses kayıtlarının MFCC öznitelik çıkarım tekniği kullanılarak veriye dönüştürülmesi yoluyla elde edilmiştir. Tasarlanan konuşmacı aksanı tanıma sistemi için 9 farklı ML sınıflandırma algoritması kullanılmıştır. Bunun yanında veri setini bağımsız olarak test edebilmek amacıyla k-katlamalı çapraz doğrulama tekniği kullanılmıştır. Bu sayede veri setini farklı sayıda parçalara bölerek analiz edildiğinde sergilediği performans gösterilmiştir. Kullanılan sınıflandırma algoritmaları ve bu algoritmalarda yapılan hiper parametre optimizasyonları açıklanmıştır. Sınıflandırma yapılarının elde ettiği başarı sonuçları değerlendirme ölçütleri kullanılarak gösterilmiştir.
Anahtar Kelimeler: Mel-frekans kepstral katsayıları, makine öğrenmesi, konuşmacı aksanı tanıma, öznitelik çıkarımı

I. INTRODUCTION

Studies on speech and speaker recognition systems have gained popularity over the past 50 years. Many studies have been done on speech and speaker recognition [1-2]. In speaker recognition, it is aimed to analyze the parametric sound information of the human voice and to recognize this information by computers at the highest possible accuracy. The first process performed in speaker recognition is to analyze the sound and converting the parametric information of the sound into data. Obtained data is passed through the predetermined filters and then converted into a text form so that the system can easily understand. Research has shown that any two people's voices are different from each other [3]. When distinguishing human voices, it is quite difficult to distinguish it acoustically. The main distinctive features used to distinguish speakers are the structure of the speaker's vocal cords, way of speaking, accent, gender, and age [4-6]. These features are extracted from speech signals using feature extraction techniques.

Corresponding Author: Ahmet Aytuğ AYRANCI, Tel: 0212 498 41 41, e-posta: ahmet.ayranci@iku.edu.tr
Submitted: 14.03.2021, **Revised:** 17.12.2021, **Accepted:** 17.12.2021

With the developing technology, the number of systems based on speech and speaker recognition is increasing day by day. The performance of speaker recognition systems is of great importance in voice-controlled systems and devices. Speech and speaker recognition systems are actively used in areas such as online banking transactions, online shopping, applications requiring database access and personal security transactions. Since these systems are critical in terms of security, accuracy is of great importance. Filters and feature extraction techniques are used on the audio signal to increase the accuracy [7-8]. Also, noise and distorting effects in the environment where the sound is obtained play a big role in successful speaker recognition.

One of the biggest impacts on the accuracy of the speaker recognition system is the correct identification of the speaker's accent. An accent is a feature that distinguishes a certain group of people from other people in terms of the way they speak. Although there have been developments in the studies conducted in recent years, the differences in the speech styles of different local groups due to social and cultural reasons cause difficulties in identifying different accents. Therefore, it is seen that a limited number of accents are used to keep the accuracy high in the speaker recognition systems. The problem of speaker recognition appears to be intrinsically a pattern recognition problem. The way speakers speak and the pattern in the choices of words they use when speaking form, the basis of recognizing the speaker's accent.

The two main tasks within speaker recognition are speaker identification and speaker verification [9-10]. Feature extraction techniques and classification methods to be used on the sound data in speaker recognition systems are of great importance. In order to obtain important information from speech, it is necessary to determine the feature extraction technique to be used first. The most widely used feature extraction techniques in the literature are Mel-Frequency Cepstral Coefficients (MFCC), Linear Prediction Coefficients (LPC), and Perceptual Linear Prediction (PLP) techniques [11-12]. The first process in speaker recognition is performing the feature extraction technique. The success performance of the method to be chosen here greatly affects the performance of other processes. The most popular feature extraction technique used in speech and speaker recognition systems is the MFCC technique. The MFCC technique works in a structure similar to the auditory system that people have. Since the MFCC processes the audio signal by estimating the frequency spectrum, the accuracy of the predicted frequency spectrum is of great importance [12]. The MFCC technique logarithmically receives components above 1000 Hz while keeping the frequency components below 1000 Hz linear to maintain important properties of the sound signal. The MFCC technique receives components above 1000 Hz logarithmically while keeping the frequency components below 1000 Hz linear to maintain important properties of the sound signal. The windowing process is applied to divide the defined signals into frames. Then, frequency components are obtained by applying the Fast Fourier Conversion (FFT) process to the windowed signals. After these processes, Mel filter bank and Discrete Cosine Transform (DCT) operations are applied. Among the feature extraction techniques, the MFCC technique appears as the highest performance technique. Apart from MFCC, PLP and LPC techniques are also commonly used feature extraction techniques in the literature.

Classification techniques are used in order to perform speaker accent recognition of the data obtained from the audio signal by applying the feature extraction technique. Speaker recognition is performed using Machine Learning (ML) and Deep Learning (DL) techniques on the data obtained by feature extraction techniques. In our study on speaker accent recognition [13], we performed performance analysis using six ML algorithms on the dataset containing six different accent information. In [14], the authors conducted an accent recognition study using the K-Nearest Neighbor (K-NN) algorithm. In [15], the verification of the speaker was carried out by using the I-vector technique. In [16] where MFCC, PLP, and LPC feature extraction techniques are used, the authors have made performance analysis on the speaker recognition system using the Support Vector Machines (SVM) classification algorithm. In [17], the performance of the classifier algorithms was tested using the Speaker Accent Recognition dataset. In this study, the data set is extended by adding the Turkish accent.

In this study, the Speaker Accent Recognition dataset from UC Irvine (UCI) ML dataset repository was used. Also, voice recordings of 37 Turkish participants were taken from the Speech Accent Archive dataset created by Steven H. Weinberg from George Mason University were used. The voice recordings of 37 speakers in the Speech Accent Archive dataset were converted into data using a 12-coefficient MFCC feature extraction technique. With the added data, the dataset was expanded by obtaining more speakers and accent numbers. The accuracy of the system has been investigated using various ML classification algorithms on the dataset. Also, the k-fold cross-validation technique was used to show each classification algorithm's performance with unbiased accuracy and reliability. Using seven different speech accents, an extensive speaker accent recognition study was carried out.

II. FEATURE EXTRACTION TECHNIQUES

Feature Extraction is the process of obtaining important information about the speech from a stationary and short part of the audio signal. Parametric information in the audio signal is obtained using feature extraction techniques. The most commonly used feature extraction techniques in the literature are MFCC, PLP, and LPC techniques. While choosing the feature extraction technique to be used, the structure of the audio signal and the noise in the environment where the sound is recorded should be considered. The performance of the feature extraction technique in speaker recognition also greatly affects the performance of other processes.

The choice of feature extraction technique has a big impact on the accuracy of speaker recognition. The reason for that, speaker recognition is based on the data obtained by the feature extraction technique. In this study, the MFCC technique is used. MFCC will be further discussed in detail in Section 2.1. PLP technique, one of the other widely used feature extraction techniques, aims to achieve high performance by using three important structures to remove unnecessary information in the audio signal. PLP feature extraction technique combines the critical bands, intensity to loudness compression, and equal loudness pre-emphasis structures in the extraction of parametric information from audio signal. PLP has been developed by taking the human auditory system as a model. The most important disadvantages of the PLP technique compared to MFCC and LPC feature extraction techniques are the lower computational speed and noise resistance. LPC feature extraction technique is designed with inspiration from the Human vocal tract. LPC evaluates the audio signal by approximating the formants, getting rid of its effects from the audio signal, and estimates the concentration and frequency of the left behind residue [18]. LPC performs better in situations where the speakers speak shortly and say the same words. The space between people's vocal cords and larynx produces a buzz. Frequency and intensity content varies according to pitch. The vocal paths are separated according to their resonant frequencies, and this is called formants. Using this technique, the positions of the formants in an audio signal are predictable by calculating the linear predictive coefficients above a sliding window and finding the crests in the spectrum of the subsequent linear prediction filter [18]. After the formants are obtained, the conversion is achieved by performing the reverse filtering process. LPC has a high performance similar to the MFCC technique in terms of computational speed, accuracy, and obtaining acoustic information of the sound.

2.1. Mel-Frequency Cepstral Coefficients

MFCC is a feature extraction technique designed based on the auditory systems of humans. MFCC converts the signal from the time domain to the frequency domain. Speech signals contain tones of varying frequencies MFCC computes these frequencies on the Mel scale. The Mel scale is approximately linear up to 1 kHz and logarithmic above the 1kHz threshold. Since the sensitivity of the human ear decreases after the 1 kHz threshold, this scaling is of great importance in sound signal extraction [11]. MFCC technique is a replication of the human auditory system intending to artificially implement the human's hearing principle to the computers. The relation between frequency of speech and Mel scale

$$Mel(f) = 2595 * log_{10}\left(1 + \frac{f}{700}\right) \quad (1)$$

can be given. It is possible to see the processes taking place in the MFCC feature extraction technique on the block diagram in Figure 1.

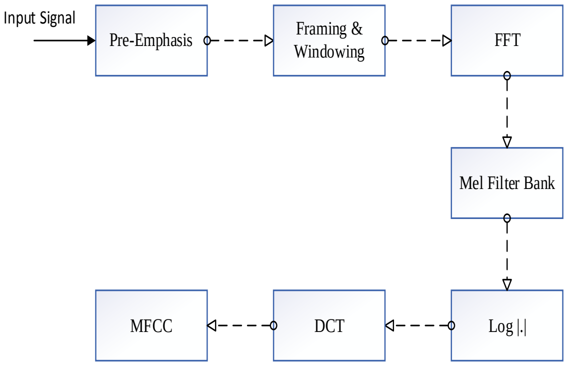

Figure 1. Block diagram of MFCC technique

Figure 1 shows the processes in the MFCC technique, respectively. In the pre-emphasis stage, the signal at high frequencies is made more pronounced. In this way, the amplitude of the high-frequency audio signal approximates the amplitude of the low-frequency audio signal and makes it suitable for comparison. This process is done by increasing the energy of the high-frequency signal. In framing, it is used to help calculate the FFT computation easily. At this stage, discontinuity is prevented. In the windowing process, each individual frame is minimized in order to prevent signal discontinuities at the beginning and end of each frame. Hamming, Hanning, Blackman, and Gauss functions are examples of commonly used windowing functions. The next step after windowing is the FFT process. Each frame is converted from the time set into a set of frequencies. This process is implemented to calculate the power spectrum. Then it is defined which frequencies are presented in which frame. One of the most important steps in the MFCC technique is the Mel filter bank. Since the human hearing ability is different in each frequency band, scaling is performed in the Mel filter bank. Since the MFCC technique is similar to the human auditory system, its sensitivity decreases as higher frequencies are reached, so it is aimed to reduce losses by taking values through a logarithmic function in the scaling process. In the last stage, the discrete cosine transformation is made. The purpose of discrete cosine transformation is to convert the Mel spectrum back to the time domain and decorrelate the filter bank energies. The result obtained is called the MFCC. One of the most important usage advantages of MFCC is its high accuracy rate in high-frequency signals. Also, MFCC has the highest performance in computational speed and capturing the sound characteristic in speech.

III. MATERIAL AND METODOLOGY

3.1. Dataset

Success performance in speaker recognition problems depends on the dataset, feature extraction techniques, and classification algorithms used. The noise in the audio file used, the variety of accents in the data set, and the selection of the feature extraction technique are of great importance in speaker accent recognition.

In this study, the data from the Speaker Accent Recognition data set taken from the UCI ML data set source was used. Also, Turkish voice recordings in the Speech accent archive dataset created by Steven H. Weinberg were converted into data with 12-coefficient MFCC and used. Voice recordings of 37 Turkish participants were taken from the Speech accent archive dataset and converted into data. Turkish voice recordings were converted into 12-coefficient MFCC data using the Praat program and added to the dataset. The number of speakers in the dataset is 367, and there are 7 accents in total.

Table 1. Dataset information

Speaker Accent	Participant
Spanish Accent	30
French Accent	30
German Accent	30
Italian Accent	30
English Accent	45
US Accent	165
Turkish Accent	37

3.2. K-Nearest Neighbors

K-Nearest Neighbor (K-NN) is one of the simplest ML algorithms based on the Supervised Learning technique. K-NN is a widely used algorithm in classification and regression problems. K-NN algorithm was developed in 1967 by T. M. Cover and P. E. Hart. Unlike other supervised learning algorithms, the K-NN classifier does not have a training phase. K-NN can be classified as a lazy learning algorithm because it does not have a specialized training phase. K-NN has a system based on memorizing rather than learning. It can be easily integrated into the classification problems. K-NN algorithm has high performance against noisy data. Because of this, it is widely used in speech and speaker recognition problems [13-14].

First, the k parameter is determined in the K-NN algorithm. This parameter is the number of neighbors closest to a given point. The distance of the new data from the existing data is calculated according to the selected k parameter. Choosing the optimal k value is important for the algorithm. In general, a large k value is more preferrable as it reduces the overall noise but there is no guarantee that algorithm performs better. The main distance functions used in the K-NN algorithm can be given as Euclidean and Manhattan distance functions. Formulations of Euclidian and Manhattan distance functions

$$\sqrt{\sum_{n=1}^{k}(x_n - y_n)^2} \tag{2}$$

$$\sum_{n=1}^{k}|x_n - y_n| \tag{3}$$

can be given. The formula given in Equation (2) shows the Euclidean distance formula. Here the x indicates the incoming value, while the y value indicates the center point. Euclidean and Manhattan distance functions can be used when the variables are continuous. In the instance of categorical variables, the Hamming distance should be used. The working method of the K-NN algorithm is given below.

- The k parameter to be used in the classification is determined. The chosen k parameter has a big impact on the accuracy of the algorithm.
- Distance calculations are made with the help of selected distance functions. The distance function selection is made according to the type of variables in the dataset used.
- K closest neighbors are selected from the related distances. The selected class is considered to be the class of the observation value expected to be estimated.

K-NN algorithm has a flexible and simple structure. It is one of the most widely used algorithms in studies on classification.

3.3. Multi-Layer Perceptron

Multi-Layer Perceptron (MLP) algorithm is used in classification and regression problems. MLP is a multi-layered feedforward neural network. MLP consists of an input layer, hidden layer (can be more than one) and an output layer.

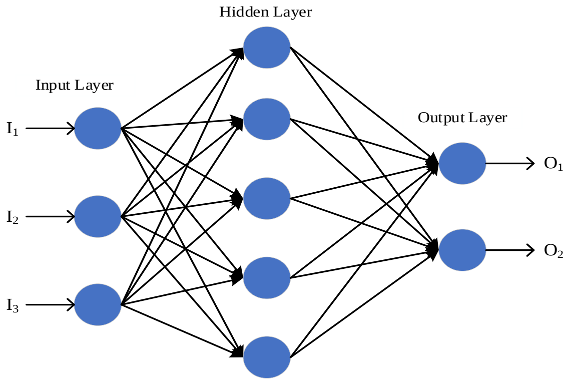

Figure 2. MLP structure

Figure 2 shows the structure of an MLP algorithm with 3 input and 2 output values. In the input layer, which is the first layer of the algorithm, data from the outside world is transmitted to the hidden layer. No operations or calculations are made in the input layer, and information is only transmitted to the hidden layer. Data from the input layer is processed in the hidden layer. There can be more than one hidden layer in the MLP structure. Information processed in the hidden layer is transmitted to the output layer. In the output layer, the data coming from the hidden layer is produced as an output.

In the MLP algorithm, there are hyper parameters that must be determined to increase the success performance. These parameters are the number of hidden layers in the algorithm, the number of iterations to be applied during the training phase, and the selection of the activation function. Determining these parameters has a big impact on accuracy. The most used activation function in the MLP algorithm is the Sigmoid function. In the MLP algorithm, all input values are multiplied and summed with the determined weight values before passing to the hidden layer. Then the obtained value is sent to the selected activation function and the output value is obtained. Accuracy of the MLP algorithm can be increased significantly with hyperparameter optimizations.

3.4. Radial Basis Function Network

Radial Basis Function (RBF) networks are an Artificial Neural Network (ANN) model developed by Moody and Darken in 1988. RBF networks are inspired by biological nerve cells. RBF networks form a special class of ANN, which consist of three layers. These layers are the input layer, hidden layer, and output layer. The feature that distinguishes RBF networks from standard ANN models is that RBF uses radial-based activation functions in the transition from the input layer to the hidden layer [19]. The hidden layer contains a number of nodes, which apply a nonlinear transformation to the input variables. The output of the RBF network is a linear combination of radial basis functions of the inputs and neuron variables. RBF networks have high accuracy in classification problems because of the simplicity and fast training of the network architecture. In the input layer, weightless connections pass inputs to the hidden layer without any processing. The hidden layer contains a number of neurons, and each neuron consists of RBF. The hidden layer applies a nonlinear transformation to the input variables, using a radial basis function, such as the Gaussian function. The variable coming from a neuron in the hidden layer is multiplied by a weight associated with the neuron. The weights are applied to the RBF function outputs as it goes to the output layer. Output in RBF networks,

$$y_n = \sum_{k=1}^{N} w_{nk}\varphi_k(x, c_k) = \sum_{k=1}^{N} w_{nk}\varphi_k \|x - c_k\|_2 \qquad (4)$$

can be expressed. In Equation (4), y indicates the output of the network, x is the input vector of the network, and φ_k indicates the radial-based activation function. N represents the number of cells in the hidden layer and w_{nk} represents the weights determined in the output layer. Optimization of two parameters is of great importance in RBF networks. These parameters are the number of neurons in the hidden layer and their output weight. Since the number of neurons that should be used in the network structure also determines the number of RBFs to be used, it can be said that it is the most important parameter.

3.5. Decision Tree

The Decision Tree (DT) algorithm is an algorithm in the supervised learning structure used in classification and regression problems. The DT algorithm has high accuracy in complex data sets or in problems where multiple classifications are required. The structure of the DT algorithm is similar to the thought structures of humans. In this way, it can be mentioned that it is a structure that is easy to understand or make inferences. Algorithm’s structure starts from the root node, goes to the inner nodes and from there to the leaf nodes, which is the final decision. The root node creates the first decision cell of the DT structure. According to the principles determined here, it is divided into inner nodes and from there into leaf nodes. The DT algorithm can be used to classify both numeric and categorical data. The creation of tree structure in numeric data sets in line with entropy values makes it difficult to understand the general structure.

Various parameters are considered when creating trees in the algorithm. While creating the DT structure, it is aimed to minimize the generalization error. The most common variants of DT algorithm used in data mining are ID3 and C4.5 algorithms. The biggest disadvantage of the DT algorithm is the problem of the over-fitting. In order to prevent this situation, restrictions or pruning can be made on parameters. Pruning the DT provides a smaller tree by removing the lower branches containing low statistics from the structure [20].

3.6. Logistic Model Tree

The Logistic Model Tree (LMT) algorithm is a classification algorithm that combines the properties of the two widely used classification algorithms. These algorithms are namely DT algorithm with the Logistic Regression (LR) algorithm. Unlike standard DT algorithms, LMT algorithm uses Logistic Regression functions on the leaf nodes. With the Logistic Regression functions used, the Logit Boost algorithm produces an LR model on each node of the tree. The nodes are then split using the c4.5 criteria [20]. LMT algorithm makes it possible to classify binary and multiple variables in the algorithm structure. Parameter optimizations and selected Logistic Regression functions greatly impacts the accuracy of the algorithm.

LMT algorithm contains the basic features of both DT and LR algorithms. However, unlike other Decision Tree algorithms, boosting is applied in the LMT algorithm to fit the LR structure to the leaves as the tree structure grows. The biggest shortcoming of the LMT is the time required to construct the tree and LR model due to its complex structure.

3.7. Random Forest

The Random Forest (RF) algorithm is a classification algorithm developed by Leo Breiman in 2001. RF can be used in both classification and regression problems. The RF algorithm is based on the combination of bagging method and random feature selection. RF is composed of multiple DT, each with the same nodes, but using different data that leads to different leaves. RF merges the decisions of multiple DT in order to find the most probable answer.

Increasing the number of trees produced in the RF structure increases the probability of obtaining a definite result. RF is considered as an ensemble learning, such that it creates more accurate results by using multiple models to come to its conclusion. This increases the accuracy of the model since RF looking at the results of many different DT structures in order to find the most probable result. [22]. The RF algorithm provides a solution to the overfitting problem, which is the biggest problem of the DT algorithm. In addition, it can provide definitive solutions to both classification and regression problems. At the same time, the RF algorithm can be used in case of missing data in the data set or data with scattered distribution. The RF structure also has high performance against noisy data, which is one of the biggest problems in the speech and speaker recognition system. Another important feature of the algorithm is that it shows the importance of attributes. The RF algorithm uses the GINI index to measure the achievements of trees and branches. The GINI index measures the success rates of the sampling given on each node. The lower the GINI index, the more homogeneous and successful that branch can be said. The algorithm can say that if the GINI index of each sub-branch is lower than the upper branch when moving towards the branches, that branch is more successful [23].

3.8. K-Star

K-Star (K*) is an entropy-based algorithm commonly used in classification problems. It has a structure based on lazy learning, such as the K-NN algorithm. The K* algorithm uses an entropic offset function to determine the similarity between attributes. K* processes the probabilities of all attributes in a category using the sum of their probabilities. It chooses the highest probability among these possibilities. The algorithm requires the optimization of a single parameter as a structure. This parameter is a global blend parameter. Global blend parameter can take values between 0 and 100. K* algorithm's function can be obtained by,

$$K^{*}(b|a) = -log_{2}P^{*}(b|a) \tag{5}$$

using the Equation (5). The P^{*} expression in this equation indicates the probability function. The K* algorithm uses the Kolmogorov distance to find shortest distance between two variables. The Kolmogorov distance or test can be used to compare two one-dimensional distributions. The Kolmogorov distance uses Cumulative Distance Functions (CDF) of the data set. Using the CDF, the distribution and maximum distances between the data are found.

3.9. Logistic Regression

LR is an ML algorithm used for classification problems. LR basically is an extension of Linear Regression model for classification problems. LR classified under Supervised Learning technique. While the Linear Regression algorithm is used for solving regression problems, and the Logistic Regression algorithm is used for solving classification problems. The most widely used LR algorithm models are binary, ordinal, and multinomial. LR can be used where the probabilities between two classes is required. In Linear Regression, the data are based on estimating the outputs by finding the best regression line. In LR, instead of fitting a regression line, we fit an "S" shaped logistic or sigmoid function, which predicts between 0 and 1. This prediction cannot exceed these limits. The sigmoid function is a mathematical function used to map the predicted values to probabilities. LR is using the concept of Maximum Likelihood estimation to estimate the accuracy. According to this estimation, the observed data should be most probable.

3.10. Naive Bayes

Naive Bayes (NB) is a probabilistic ML algorithm based on the Bayes Theorem. Bayes Theorem is a simple mathematical formula used for calculating conditional probabilities. NB Algorithm is one of the simplest Classification algorithms which helps in building fast ML models that can make quick predictions. The NB algorithm classifies the data submitted to the system with probability calculations. NB algorithm has a flexible use because it requires a small amount of data for training. In studies on speaker and speaker accent recognition, it is seen that the NB classification algorithm has a low performance [13]. Considering the results obtained in this study on speaker accent recognition, it is seen that the algorithm with the lowest accuracy is NB.

3.11. K-Fold Cross Validation

The goal in the K-Fold cross-validation structure is to be able to create unbiased observation sets. Dataset is divided into k number of parts and the training set is tested for all observation sets. While the dataset is divided into k parts, all parts are expected to be of equal size and quality. In K-fold cross-validation structure, data is divided into k equal-sized folds, then the classifier was trained using the k-1 folds and tested on the remaining partition. This process is repeated k times [24]. Using the K-fold cross-validation technique, both the performance of the classification structure and the performance change according to the k parameter can be seen [25]. K-fold cross-validation shows the prediction accuracy and usability of the classifier methods. K-fold cross-validation shows unbiased prediction accuracy and generalization performance of a classifier structure. In this study, the 10-fold cross-validation method was used in order to see the performance of algorithms with unbiased.

IV. RESULTS AND DISCUSSION

Performance tests will be conducted by using 9 ML classification algorithms on the dataset. The performance of the algorithms explained through the evaluation criteria. In this section, their performance metrics will be explained first. Then the obtained algorithm results will be given.

4.1. Performance setrics

In this section, the evaluation criteria used to compare the performance of the classification algorithms will be explained. Information will be given about the criteria that show the accuracy performance of the algorithms and the criteria that show the error performance. The main parameters used to calculate evaluation criteria are True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN).

4.1.1. Accuracy

It is the basic evaluation criterion used to see the performance of the algorithms used in classification and regression problems. Accuracy is the most basic evaluation criterion that specifies the performance of classification algorithms. The accuracy rate measures the percentage of the number of data that the algorithm correctly estimates in the total data. To calculate the accuracy rate,

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{6}$$

expression can be used. It is used to determine the algorithm's performance across classes and on the entire data set.

4.1.2. Precision

Precision shows the success rate within positive estimates. The accuracy value specifically indicates the importance of TN values. Precision value is of great importance when determining the performance of the algorithm. Precision value,

$$Precision = \frac{TP}{TP+TN} \tag{7}$$

can be calculated using Equation (7).

4.1.3. Recall

The recall parameter measures how many of the parameters that need to be positively estimated are correctly predicted. Recall criteria are used to examine the effect of positively predicted results. The formula

$$Recall = \frac{TP}{TP+FN} \tag{8}$$

is used for the sensitivity account.

4.1.4. Kappa statistics

Kappa statistics or coefficient is a criterion used in multiclass classification problems. Kappa Statistics are used because the accuracy, precision, and sensitivity criteria do not fully show the performance in multi algorithm cases. When examining Kappa Statistics, the confusion matrix should also be considered. Because of the observed accuracy and expected accuracy parameters should be used when calculating Kappa statistics confusion matrix should be given. The Kappa Statistics are always 1 or less than 1. In cases where kappa statistics are less than 0, it can be said that the classification structure has failed. Kappa Statistics, which has a more complex structure, can be shown as the most accurate evaluation criteria among classification structures. In order to calculate Kappa Statistics, two different probability calculations are required. These possibilities are observed probability $pr(o)$ and expected probability $pr(e)$. Kappa Statistics using these probability values,

$$\kappa = \frac{pr(o)-pr(e)}{1-pr(e)} \tag{9}$$

expressed in the form [26].

4.1.5. *F-Score*

F-score is a performance metric that compares precision and sensitivity together. Together with the Kappa Statistics, it is the most useful criteria for performance evaluation when there is an imbalance between classes in the dataset. To calculate F-score;

$$F - Score = 2 * \frac{Precision*Recall}{Precision+Recall} \qquad (10)$$

Equation (10) is used.

4.1.6. *Mean Absolute Error*

Mean Absolute Error (MAE) is a statistical error measure used to compare classification structures. MAE basically shows the absolute difference between two continuous variables. MAE performance metric displays the absolute distance the values in the dataset and the regression line. Because it is an easy-to-interpret criterion, it is often used to indicate performance in classification and regression problems. MAE criteria are measured regardless of the direction of errors in the forecast. MAE is a linear criterion where all individual errors are weighted equally on average. The formula,

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|e_i| \qquad (11)$$

is used to calculate the MAE criteria.

4.1.7. *Root Mean Square Error*

Root Mean Square Error (RMSE) is a widely used evaluation criteria in classification algorithms. It defines the standard deviation of difference between estimated values and true values. RMSE is also a second degree (quadratic) criteria to indicate the standard deviation of estimate errors. RMSE criteria points out the data density around regression line. As RMSE value gets close to 0, it shows that classification structure performs better. To calculate the RMSE criteria;

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n}} = \sqrt{MSE} \qquad (12)$$

formulation can be used. The MSE parameter represents Mean Square Error. RMSE value can be calculated via taking square root of MSE. MSE criteria is similar to RMSE criteria and defines the distance of a series variables to regression line.

4.2. Results

Advised classification structures in created data sets are tested by Weikato Environment for Knowledge Analysis (WEKA) program. Data set split into two parts as %70 education and %30 test and tested in 9 different classification algorithms. In used classification algorithms, various hyperparameter optimizations applied to achieve better success ratio. For example, learning speed set to 0.35 and momentum set to 0.2 for MLP structure. In K* algorithm, when global blend parameter set to 30, highest success ratio achieved. When hyper parameter optimizations like this used in classification algorithms, accuracy in designed structure's success increased.

Table 2. Evaluation criteria of algorithms

Algorithms	Accuracy	Precision	Recall
MLP	**89.1**	**89.6**	**89.1**
RBF	78.18	77	78.2
K-NN	**88.2**	**89.5**	**88.2**
K*	84.55	85.6	84.5
DT	70	70.4	70
LMT	72.8	73.6	72.7
RF	81.8	83.3	81.6
LR	69.1	69	69.2
NB	57.3	73	57.3

Table 3. Kappa Statistics and F-Score of algorithms

Algorithms	Kappa Statistics	F-Score
MLP	**84**	**88.5**
RBF	67.45	76.7
K-NN	**83.35**	**88.6**
K*	77.9	84.6
DT	56.2	69.5
LMT	58.3	71.3
RF	72.7	81.4
LR	55.4	68.5
NB	47.9	58.5

The performance metrics of the algorithms are shown in table 2 and Table 3. When the tables are examined, it is seen that the algorithms that provide the highest performance are MLP and K-NN algorithms. The classification structures are expected to achieve the highest possible values, as the above evaluation criteria show the performance of algorithms.

Table 4. Error metrics of algorithms

Algorithms	MAE	RMSE
MLP	**5.3**	**17**
RBF	7.3	24
K-NN	**3.95**	**18.2**
K*	6.5	19.1
DT	10.2	28.2
LMT	10.15	22
RF	11.5	21.2
LR	9.2	24.3
NB	13.2	30.1

Table 4 shows the evaluation criteria of classification algorithms based on error performance. When the two criteria are evaluated together, it is seen that the algorithms that provide the highest error performance are MLP and K-NN. Algorithms with the lowest error performance appear to be DT and NB algorithms.

In Figure 3, accuracy, precision and recall performance metrics of algorithms are shown. As can be seen in the figure, MLP and K-NN algorithms have shown the highest performance. Hyperparameter optimizations made in MLP, K-NN, and K* algorithms affected the algorithms' high performance.

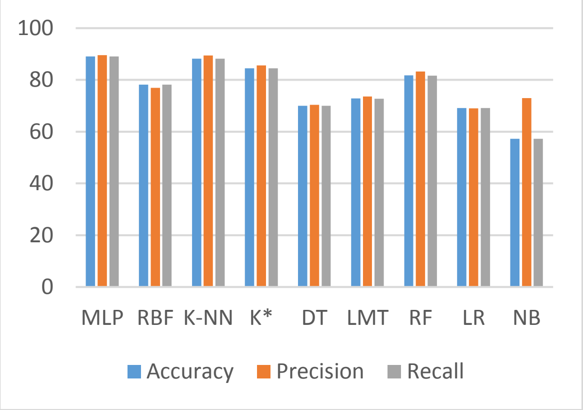

Figure 3. Accuracy, precision, and recall values

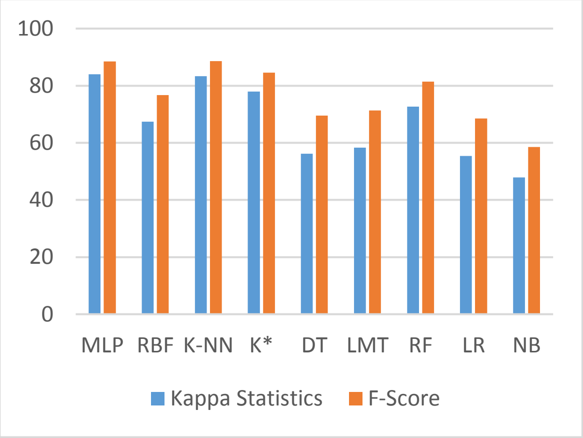

Figure 4. Kappa statistics and F-score parameters

In Figure 4, Kappa Statistics and F-Score results are shown through graphs.

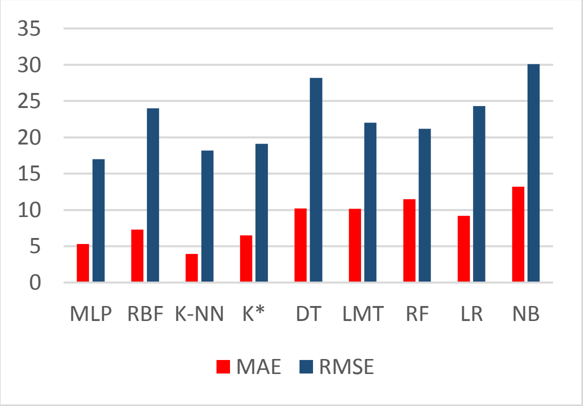

Figure 5. MAE and RMSE values of algorithms

In Figure 5, the error performance results obtained for the algorithms are shown in the graphs. Evaluation criteria are of great importance when comparing the performances of the classification structures. In addition to the evaluation criteria, a confusion matrix is used to show the performance of the classification structures and the accuracy of each class in the dataset. The confusion matrices of the classification algorithms are given below. The confusion matrices of the algorithms used are given in the tables below. The letters shown in the table (a = ES, b = FR, c = GE, d = IT, e = UK, f = US, g = TR) indicates the accents of speakers.

Table 5. Confusion matrices of algorithms.

MLP	a	b	c	d	e	f	g
a	7	0	0	0	0	2	0
b	1	9	0	0	0	0	0
c	0	0	6	0	0	1	0
d	0	0	1	3	2	1	0
e	0	0	0	0	12	2	0
f	1	0	0	0	1	54	0
g	0	0	0	0	0	0	7

a) MLP

RBF	a	b	c	d	e	f	g
a	7	0	0	0	0	2	0
b	0	2	0	1	2	5	0
c	0	0	4	0	0	3	0
d	0	0	0	6	1	0	0
e	1	0	0	0	10	3	0
f	0	2	1	0	3	50	0
g	0	0	0	0	0	0	7

b) RBF

KNN	a	b	c	d	e	f	g
a	8	0	0	0	0	1	0
b	0	8	1	0	1	0	0
c	0	0	6	1	0	0	0
d	0	0	1	6	0	0	0
e	0	0	0	0	12	2	0
f	0	1	2	0	3	50	0
g	0	0	0	0	0	0	7

c) K-NN

K*	a	b	c	d	e	f	g
a	8	0	0	0	0	1	0
b	0	7	1	1	0	1	0
c	0	0	6	0	0	1	0
d	0	0	1	6	0	0	0
e	0	0	1	1	9	3	0
f	1	0	1	0	4	50	0
g	0	0	0	0	0	0	7

d) K*

DT	a	b	c	d	e	f	g
a	5	0	0	0	1	3	0
b	0	4	0	2	1	3	0
c	0	0	6	0	0	1	0
d	0	1	2	3	0	1	0
e	1	0	1	1	6	5	0
f	1	1	3	2	1	47	1
g	0	0	0	0	1	0	6

e) DT

LMT	a	b	c	d	e	f	g
a	6	0	0	0	0	3	0
b	0	1	3	1	1	4	0
c	0	0	4	0	0	3	0
d	0	0	2	4	0	1	0
e	0	0	0	1	9	4	0
f	0	2	3	0	0	51	0
g	0	0	0	0	0	2	5

f) LMT

RF	a	b	c	d	e	f	g
a	6	0	0	0	0	3	0
b	0	6	0	3	0	1	0
c	0	0	4	0	0	3	0
d	0	0	2	5	0	0	0
e	0	0	0	0	9	5	0
f	0	0	0	1	2	53	0
g	0	0	0	0	0	0	7

g) RF

LR	a	b	c	d	e	f	g
a	6	0	0	0	0	3	0
b	0	2	2	3	1	2	0
c	0	0	4	0	0	3	0
d	0	0	0	4	2	1	0
e	0	0	0	2	8	4	0
f	4	3	4	0	0	45	0
g	0	0	0	0	0	0	7

h) LR

NB	a	b	c	d	e	f	g
a	7	2	0	0	0	0	0
b	1	2	0	7	0	0	0
c	0	0	7	0	0	0	0
d	0	0	0	6	1	0	0
e	0	1	1	1	10	1	0
f	7	9	3	6	6	24	1
g	0	0	0	0	0	0	7

i) NB

The confusion matrices of the classification algorithms are given in Table 5. The letters in the matrices indicate the Spanish accent, the French accent, the German accent, the Italian accent, the British accent, the American accent, and the Turkish accent, respectively. Successful predictions made by algorithms for each accent can be seen over tables. In Table 2, the most successful results are obtained in MLP and K-NN algorithms. When the confusion matrix tables of these algorithms are examined (Tables 5. a, and 5. c), it is seen that these algorithms have shown more accurate results than other algorithms, and their margin of error is low. By examining the confusion matrix, it is possible to see which features of the dataset causes errors in the classification algorithm and the distribution of these errors. The accuracy of each class in the dataset also can be seen for all algorithms via the confusion matrix.

To see the performance of algorithms unbiased, k-fold cross-validation technique was used in the study. Here, the k number was chosen as 10 to obtain the most unbiased results. Table 6 shows the results obtained for 10-fold cross-validation in each algorithm used.

Table 6. Performance metrics for 10-fold cross-validation

Algorithms	Accuracy	Kappa Statistics	F-Score
MLP	**82.8**	**76.8**	**82.4**
RBF	81.15	74.5	81.1
K-NN	**82.78**	**77.1**	**83**
K*	78.15	70.8	78.8
DT	63.7	51.5	63.7
LMT	72.95	63.5	73.1
RF	82	74.9	81.5
LR	74.6	65.6	74.5
NB	60.4	52.1	60.6

When 10-fold cross-validation is used, the evaluation criterion values obtained in the algorithms are given in Table 6. The algorithms with the highest performance also appear to be MLP and K-NN algorithms.

V. CONCLUSION

A study of speaker accent recognition was conducted using the MFCC feature extraction technique and ML classification algorithms. In the dataset, a total of 7 accent data provided for 367 speakers. ML classification algorithms are used in this system to test the dataset's performance. To achieve the optimum performance in the system, hyperparameter optimizations are applied in classification algorithms. The performances of the algorithms are shown in tables and graphs using evaluation criteria.

In this study, the data set was expanded compared to [13] and different ML algorithms were added. It is seen that higher performance is obtained in the study with the extended data set. MLP and K-NN were the highest performing algorithms among the ML classification algorithms that used. The algorithms that perform the worst accuracy are DT and NB algorithms. This was valid both when the data set was used for 70% training and 30% testing and when the 10-fold cross validation technique was used. It has been observed that algorithms with the high noise resistance attribute performs better performance.

REFERENCES

[1] Van Leeuwen D. A. Martin A. F., Przybocki M. A., and Bouten J. S., (2006). NIST and TNO-NFI evaluations of automatic speaker recognition. Comput. Speech Lang., vol. 20, pp. 128–158.

[2] Furui, S. (1970). 50 Years of Progress in Speech and Speaker Recognition Research.

[3] Kinnunen T. and Li H. (2010). An overview of text-independent speaker recognition: From features to supervectors, Speech communication, vol. 52, no. 1, pp. 12–40.

[4] Nakagawa S., Wang L. and Ohtsuka S. (2012). Speaker identification and verification by combining MFCC and phase information. *IEEE Trans. Audio Speech Lang. Process.*, vol. 20, no. 4, pp. 1085-1095.

[5] Faria, A. (2005). Accent classification for speech recognition *proceedings of the Second Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI '05).*

[6] Turner C. and Joseph A. (2015). A wavelet packet and mel-frequency cepstral coefficients-based feature extraction method for speaker identification Procedia Computer Science, 61, pp. 416-421.

[7] De-la-Calle-Silos F. and Stern R. M. (2017). Synchrony-Based Feature Extraction for Robust Automatic Speech Recognition. in IEEE Signal Processing Letters, vol. 24, no. 8, pp. 1158-1162.

[8] Ranjan R. and Thakur A. (2019). Analysis of feature extraction techniques for speech recognition system, *International Journal of Innovative Technology and Exploring Engineering,* vol. 8, no. 7C2, pp. 197-200.

[9] Reynolds D. A. and Rose R. C. (1995). Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. *IEEE Trans. Speech and Audio Processing,* vol. 3, no. 1, pp. 72-83.

[10] Campbell J. P. (1997). Speaker recognition: A tutorial. Proc. IEEE, vol. 85, no. 9, pp. 1437–1462.

[11] Dave N. (2013). Feature extraction methods LPC PLP and MFCC in speech recognition *Int. J. for Advance Research in Eng. and Technology,* vol. I, no. 6, pp. 1-4.

[12] Harris F. (1998). On the use of windows for harmonic analysis with the discrete Fourier transform. Proceedings of the IEEE, vol. 66, no. 1, pp. 51-84.

[13] Ayrancı A. A., Atay S. and Yıldırım T. (2020). Speaker Accent Recognition Using Machine Learning Algorithms. *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)* Istanbul, Turkey, pp. 1-6.

[14] Widyowaty D. S. and Sunyoto A. (2020). Accent Recognition by Native Language Using Mel-Frequency Cepstral Coefficient and K-Nearest Neighbor. *2020 3rd International Conference on Information and Communications Technology (ICOIACT)* Yogyakarta, Indonesia, pp. 314-318.

[15] Alam M. J.,Kinnunen T., Kenny P., Ouellet P. and O'Shaughnessy D. (2011). Multi-taper MFCC features for speaker verification using I-vectors. *2011 IEEE Workshop on Automatic Speech Recognition & Understanding,* Waikoloa, HI, pp. 547-552.

[16] Aslan Z. and Akın M. (2019). Performing accurate speaker recognition by use of SVM and cepstral features. The International Journal of Energy and Engineering Sciences, vol. 3, no. 2, pp. 16-25.

[17] Okkan U., Dalkılıç H. (2012). Radyal Tabanli Yapay Sinir Aglari ile Kemer Barajı Aylık Akımlarının Modellenmesi. İMO Teknik Dergi, 379, 5957-5966.

[18] S. A. Alim and N. K. A. Rashid. (2018). Some commonly used speech feature extraction algorithms. in From Natural to Artificial Intelligence-Algorithms and Applications, IntechOpen.

[19] Onan A. (2015). Şirket İflaslarının Tahminlenmesinde Karar Ağacı Algoritmalarının Karşılaştırmalı Başarım Analizi. Bilişim Teknolojileri Dergisi, vol. 8, no. 1, pp. 0.

[20] Landwehr, N., M. Hall, and E. Frank, Logistic model trees. Machine learning, (2005). 59(1-2): p. 161-205.

[21] Akar Ö. and Güngör O. (2012). Rastgele Orman algoritması kullanılarak çok bantlı görüntülerin sınıflandırılması, Jeodezi ve Jeoinformasyon Dergisi, no. 106, pp. 139-146.

[22] Erdem F., Derinpınar M., Nasırzadehdızajı R., Oy S., Şeker D. and Bayram B. (2018). Rastgele Orman Yöntemi Kullanılarak Kıyı Çizgisi Çıkarımı İstanbul Örneği. Geomatik, vol. 3, no. 2, pp. 100-107.

[23] Kartal C. (2020). Modeling Bitcoin with K-Star Algorithm. bmij, vol. 8, no. 1, pp. 213-231.

[24] Mutlu, A. Y. and Yucel, O. (2018). An artificial intelligence based approach to predicting syngas composition for downdraft biomass gasification. Energy, vol. 165, pp. 895–901.

[25] Elmaz F. , Yücel Ö. and Mutlu A. (2020). Machine learning based approach for predicting of higher heating values of solid fuels using proximity and ultimate analysis", *International Journal of Advances in Engineering and Pure Sciences,* vol. 32, no. 2, pp. 145-151, Jun. 2020.

[26] Landis J. R. and Koch G. G. (1977). The measurement of observer agreement for categorical data. Biometrics, 33(1):159.

[27] Kılıç, S. (2015). Kappa Testi. Journal of Mood Disorders, 5(3), 142-144.