Research Article

Luis David Huerta-Hernández, Nayeli Joaquinita Meléndez-Acosta, David Ernesto Troncoso-Romero, Julio César Ramírez-Pacheco, and Jose Antonio León-Borges*

Speech emotion recognition using long-term average spectrum

https://doi.org/10.1515/comp-2025-0023
received September 24, 2024; accepted January 16, 2025

Abstract: Automatic speech emotion recognition has become an important research subject in the area of speech signal processing. The performance of classification algorithms depends on the features extracted from speech. In this work, a new framework for emotion recognition is proposed based on the long-term average spectrum (LTAS). Our framework is evaluated through a comparative study, where classifiers such as artificial neural network, K-nearest neighbours, logistic regression, Bayesian algorithms, tree-based logistics, and support vector machine were used. The framework was experimentally tested using the well-known Toronto Emotional Speech Set database, and the results were compared against state-of-the-art alternatives, using mel frequency cepstral coefficients, filter bank energies, and chroma coefficient speech coding, on this database. Comparative experiments showed that the use of LTAS achieved higher performance, with accuracies of 96–99% in terms of correct classification of speech emotion, compared with the best performance of 97% for the state-of-the-art alternatives. Different sampling frequencies were used to extract LTAS, and the classifiers were tested individually. The main contribution of this work is to demonstrate that the new framework using LTAS significantly reduces the number of parameters down to 87.5 values per s (approximately), as opposed to the 1,200 values used in the best-performing state-of-the-art alternatives; this means that the process of feature extraction is significantly reduced and the performance in terms of correct classification is improved.

*** Corresponding author: Jose Antonio León-Borges,** Universidad Autónoma del Estado de Quintana Roo, Cancún, Quintana Roo, 77519, Mexico, e-mail: jleon@uqroo.edu.mx
Luis David Huerta-Hernández: Universidad del Istmo, Ixtepex, Oaxaca, 70110, Mexico, e-mail: luisdh2@bianni.unistmo.edu.mx
Nayeli Joaquinita Meléndez-Acosta: Universidad del Istmo, Ixtepex, Oaxaca, 70110, Mexico
David Ernesto Troncoso-Romero, Julio César Ramírez-Pacheco: Universidad Autónoma del Estado de Quintana Roo, Cancún, Quintana Roo, 77519, Mexico

Keywords: speech emotion recognition, signal processing, long-term average spectrum, classification

1 Introduction

Automatic speech emotion recognition (ASER) is a challenging task, due to the gap between real human emotions and acoustic features [1]. ASER plays a very important role in human–computer interaction (HCI), as it provides important psychological information about a speaker, thus enabling devices to recognise a person's emotions through conversation and then to make relevant decisions about them. Emotion recognition also lies at the centre of affective computing and has had many applications for several decades; its applications have been visualised in the study by Picard [2]. Emotion recognition is used in areas such as multimedia applications, health care, and HCI [3]. Research has shown that an absence of emotion causes discomfort for a human when communicating with a computer [4], suggesting that the use of emotions in human–computer interfaces could be desirable. In addition, studies of the influence of certain products on clients' feelings can determine whether the product will be bought or rejected [5]. Also, studies of products influencing client feeling, could determine if the product is bought or rejected [5]. In the area of education, affective computer applications could improve and prepare the student's mental state for learning [6–8].

From experiments with emotions, it has been suggested that an extremely positive mood is not beneficial for learning and that a slightly negative emotional state promotes critical thinking [9]. In areas such as psychology, emotion recognition could help psychologists to detect emotions in people with expressive difficulties [10]. Applications in the domain of rehabilitation, where emotional comprehension of the patients is crucial, could result in shorter and more successful recoveries. Researchers in psychology and neuroscience have been interested in the benefits of emotions, and in this context, the use of automated emotion recognition can be helpful [11].

An ASER system with low complexity in terms of implementation and a quick response time could be desirable in the areas discussed earlier. Feature selection poses an important challenge for ASER, due to its strong influence on the performance of the whole system, and there are no single features, frameworks, methods, or algorithms that yield high and stable performance in all scenarios of speech emotion classification. Research and experimentation in this area is therefore valuable.

Speech emotion classification results have been reported in several studies. In the study by Jiang et al. [1], the interactive emotional dyadic motion capture dataset was used, and an accuracy of 64% was obtained for correct classifications. Kerkeni et al. [12] used the Berlin and Spanish databases and combined empirical mode decomposition with the Teager-Kaiser energy operator to achieve an accuracy of 91.16%. Their experiments involved a set of features such as energy cepstral coefficients, frequency-weighted energy cepstral coefficients, and mel frequency cepstral coefficients based on the reconstructed signal.

Praseetha and Vadivel used the TESS database and a deep neural network (DNN) to obtain an accuracy of 89.58% and used a recurrent neural network (RNN), known as a gated recurrent unit (GRU), for speech emotion recognition with an accuracy of 95.82%, using the features of Mel frequency cepstral coefficient (MFCC) and delta MFCC [13]. Two other studies by Shaw and Saxena in 2016 [14] and Palo and Chandra in 2015 [15] used MFCC. Karimi and Sedaaghi [16] and Emerich and Lupu [17] used time-domain based methods. A discrete wavelet transform was also used by Emerich and Lupu, and their results indicated an accuracy of 96.57%. Perceptual linear prediction (PLP), linear prediction coefficient (LPC), linear prediction cepstral coefficient (LPCC), and MFCC were used by Palo and Chandra, who reported an accuracy of 80%. Nanavare [18] and Bastug [19] also experimented with classification for speech emotions using MFCC, as this feature is typically employed for speech processing.

Most existing methods for ASER use MFCC as the main feature, as this leads to the extraction of detailed information from speech; however, the number of parameters (values) involved in these classification methods is relatively high. The values for codification of the speech scheme must be processed by filters and classifiers, which affects the time required for value extraction, filtering, and training, as well as the precision of the classification [20,21], among other disadvantages. Efforts have been made to reduce the number of speech parameters [22,23].

Gambhir et al. [24] worked with the Hindi language, and considered 27,145 speech keywords developed by Tata Institute of Fundamental Research and 23,664 of 1-s utterances English speech commands, using Google TensorFlow and Artificial Intelligence Yourself English Speech Commands, which had not been explored and examined well on AVR systems [24]. In their article, they presented a three-layered two-dimensional sequential convolutional neural architecture (Sequential Conv2D) as an end-to-end system that could instantaneously exploit speech signal spectral and temporal structures. They trained and tested their model on different cepstral features such as frequency and time variant-mel-filters, gamma-tone filter cepstral quantities, bark-filter band coefficients, and spectrogram features of speech structures. The performance of convolutional layers trained on spectrograms was reported to give an accuracy of 91.60%, better than for other cepstral feature labels for English speech. The same model achieved an accuracy of 69.65% for Hindi audio words, where bark-frequency cepstral coefficients features outperformed spectrogram features.

Many other works in the field of speech emotion classification have been proposed; however, the high number of features involved and their impacts in various ways on computational cost mean that these methods are unsuitable for emotion classification, primarily when the response time is crucial.

In this article, an ASER framework based on LTAS is proposed, its performance is tested, and a comparative study is conducted with current methods in the literature, using the Toronto Emotional Speech Set (TESS) database. The experimental study was performed with seven emotions (classes): anger, disgust, fear, happiness, pleasantness, surprise, sadness, and neutrality. Our framework based on LTAS was tested against a multilayer perceptron (MLP) neural network, k-nearest neighbours (KNN), sequential minimal optimisation (SMO), and a variety of tree-based, logistic, and Bayesian algorithms. The main aims of this work were to experiment with alternative speech features and to reduce the number of features and their respective parameters, to make this approach suitable for the high performance requirements of speech emotion classification. This article is organised as follows: Section 2 reviews the main features used for speech emotion recognition. In Section 3, we describe the methods and frameworks employed in this study. Section 4 explains the speech codification process, based on frequencies and features in the time domain that are commonly used in ASER. Section 5 presents the experimental results and the methods used. The conclusions are drawn in Section 6, with suggestions for further work.

2 Speech features

In this section, we review the speech codification process based on the frequencies and features of the time domain that are commonly used in emotion recognition. The energy, zero crossing rate, and pitch are measures commonly used in the time domain, whereas in the frequency domain, the most widely used are MFCC, LPC, LTAS, and mel filter banks.

2.1 Energy

The energy is obtained as the sum of squares of the signal amplitudes, and it is proportional to Pa^2s [17,25] because $x(t)$ is the amplitude of sound, given in Pa. Then, the energy is defined as follows:

$$f = \sum_{-\infty}^{\infty} \cdot x^2(t). \tag{1}$$

This feature has been used for vowel recognition from speech signals, phoneme speech segmentation [26], and the detection of voice presence in high-quality speech signals.

2.2 Fundamental frequency

The fundamental frequency (F_0) is the vibratory frequency of the vocal cords. The number of air pressure oscillations per second determines the F_0. The pitch is related to F_0, as it is how F_0 is perceived by our ears. The F_0 is usually used for tasks such as gender identification and as an important indicator of emotional status.

2.3 Mel spectra

Mel spectra are obtained by passing a speech signal through a triangular filter bank, where the result is given as a vector of spectra. Each spectrum in the vector is obtained by an individual filter, where the size of the vector is the same as the number of filters in the bank (Figure 1).

The axes of the triangular filters are distributed according to the nonlinear mel scale proposed by Stevens and Volkman [27], where the borders of the filters lie on adjacent frequency axes. These authors proposed that the perception level with respect to a frequency that is heard follows a logarithmic scale expressed by the equation:

$$F(f) = 295 \cdot \log_{10}\left(1 + \frac{f}{700}\right). \tag{2}$$

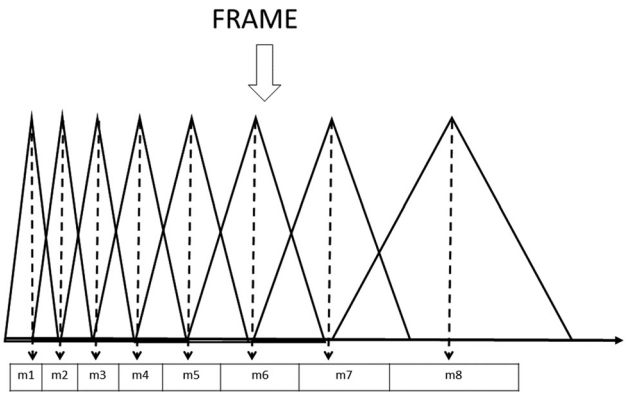

Figure 1: Extraction of a spectrum through mel filter banks.

This speech codification has been used in various types of speech research, such as speaker recognition and speech segmentation [28].

2.4 Linear predictive coding (LPC)

LPC is a scheme for representing voice production. The LPC models a filter, through which sound source passes. The filter represents a function for obtaining speech parameters such as pitch, formants, and vocal tract [29]. It is helpful in terms of representing signals in a compressed way for net transmissions, thereby removing redundancy. The model is also known as linear prediction (LP), where a speech signal is approximated as a linear combination of its previous values [30].

The MFCC measures the spectral variations of frequencies, thus enabling the treatment of the frequencies in the time domain [31] as the cepstrum is a periodicity measure of frequency. This approach to speech codification has been used in many applications, such as the detection and classification of infants' cries [32,33].

2.5 MFCCs

MFCC is the most typical method used to extract spectral information from audio data. MFCCs are based on the mel scale in the frequency domain, which follows the distribution of the human ear. This codification scheme for audio has been applied to a wide variety of studies into speech processing due to its robustness in regard to speaker and recording variability [30].

MFCC extraction starts by applying a pre-emphasis filter to emphasise the higher frequencies, as these frequencies are reduced in the recording process; at the same time, some effects from the glottal source are also reduced. The spectral slope is increased by 6 dB/octave **over** the frequency F as the input parameter. A Hamming window is used to reduce spectral distortions by splitting the continual signal into frames of 20 or 30 ms (typically), overlapping by 10 ms [34].

The MFCC features are derived from the fast Fourier transform (FFT) spectral magnitudes passed through the mel filters bank, and the logarithm of the energy on each filter is computed before applying a cosine discrete transform to produce the MFCC feature vectors.

2.6 Long-term average spectrum (LTAS)

LTAS represents the averaged spectral information on the vocal tract over the frequencies. The process of extracting LTAS from a speech signal involves applying framing with overlapping and computing its spectral power and then obtaining the spectral average. A Hamming window is then applied to each frame to smooth the edges, and an FFT is applied to these frames [35]. The LTAS is obtained using a frequency step (bin width) that ranges from zero to the maximum frequency of the speech signal. At each step, an averaged spectrum is computed. The size of the LTAS vector is the same for all signals with the same sampling frequency. LTAS represents the power spectral density in dB/Hz and is frequently expressed in logarithmic form as follows:

$$\mathrm{PSD}_{\mathrm{db}}(f) = 10 \cdot \log_{10}\left\{\frac{\mathrm{PSD}(f)}{P_{\mathrm{ref}}^2}\right\}, \tag{3}$$

where $P_{\mathrm{ref}} = 2 \cdot 10^{-5}$ Pa. Then, this logarithmic power spectral density is the quantity stored in an LTAS [25]. This power spectral density can be calculated from the sound complex spectrum $x(t)$ in the time range $T = t_2 - t_1$, measured in $\mathrm{Pa}^2/\mathrm{Hz}$:

$$\mathrm{PSD} \equiv \frac{2|X(f)|^2}{T}. \tag{4}$$

The average sound power is obtained in the range time (t_1, t_2) as follows:

$$\int_0^F \mathrm{PSD}(f)\mathrm{d}f = \frac{1}{T}\int_{t_1}^{t_2} |X(t)^2| \cdot \mathrm{d}t. \tag{5}$$

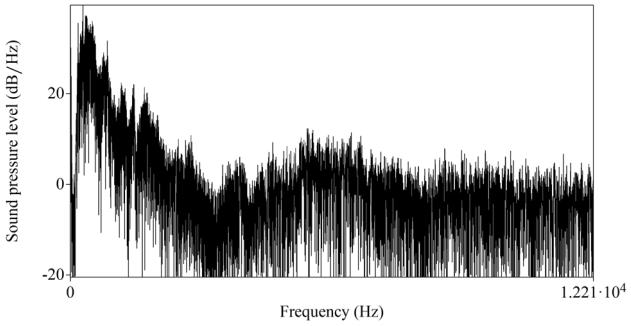

Figure 2: Spectrum for the sentence “Say the word wire.”

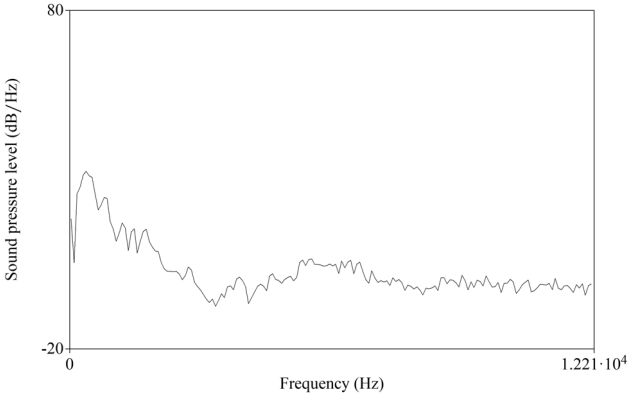

Figure 3: LTAS for the sentence “Say the word wire.”

LTAS is the average of the spectrum over a long speech signal and is used to describe the resonance characteristics of a speaker; in general, it is useful for recognition of a speaker [35], including gender, age, and diseases [36] and forensic usage [37]. In general, LTAS encodes speech signals with the voice quality [38].

In order to demonstrate the simplicity of LTAS representation, the spectrum and LTAS were extracted from the sentence “Say the word ‘wire”’, spoken by a woman, and both kinds of signals were graphed. The spectrum shows a high level of variability in its values over the frequencies (Figure 2), and a summarised version of the spectrum is therefore used, such as LTAS (Figure 3).

3 Speech emotion recognition

Several features are used in the speech recognition process, such as delta MFCC and MFCC [13]; MFCC, LPC, LPCC, and PLP [15]; and MFCC, zero crossing rate, energy, and chroma coefficients [39], among others. In this section, we review the different speech measures and coding used in

the literature and describe the usage of different measures and speech codification processes.

3.1 MFCC, LPC, LPCC, and PLP for emotion recognition

MFCC, LPC, and PLP were described and compared in terms of their individual effectiveness for emotion recognition by Palo and Chandra [15] using three and four classes (boredom, anger, sadness, and surprise). The results indicated that an MFCC-based method could achieve the best accuracy of 80.00–83.20% followed by a method based on PLP speech codification (70–74.3% accuracy), and finally, a framework based on LPC (48.60–56.20% accuracy). The authors concluded that methods based on MFCC gave the best performance.

3.2 Excitation source analysis with MFCC

Pravena and Govind [40] experimented with the IITKGP-SESC and EmoDb emotional databases. The speech codification used was MFCC, and a Gaussian mixture model (GMM) was used as a recognition system. With EmoDb, four classes were considered (anger, happiness, boredom, and neutrality), and 80% of the total utterances were tested. The best performance in terms of emotion recognition was obtained using 256 Gaussian mixtures, where the average accuracy was 73.28% for the classification of these five emotions. In this study, 39 MFCC coefficients were extracted from 20 ms frames with an overlap of 10 ms, and information on the MFCC velocity and acceleration coefficients of each emotion utterance was obtained.

3.3 Mixed time- and frequency-domain features

In work by Karimi and Sedaaghi [16] and Sundarprasad [39], time and frequency-domain features were used. The statistical features of pitch, energy, MFCC, LPC, and PLP gave performance of 82.6% using Bayes classification on the EmoDB database [16], while GMM and KNN gave accuracies of between 79.52 and 75.61%, respectively. The zero-crossing rate, spectral values, MFCC, and chroma coefficients were used by Sundarprasad et al. [39]. The implementation and results of those works will be presented in Section 5.

4 Methodology

In this section, we briefly describe the features extracted from the speech signals, and the classification algorithms, database, and tools used in the experiments. PRAAT and WEKA software packages were used in our experiments, with classifiers such as MLP, KNN, logistic, simple logistic (SL), SMO, random forest (RF), random tree (RT), J45 tree, Bayesian network (BN), and Naïve Bayes (NB).

4.1 Database

The signals considered in our experiments were obtained from the TESS database [41], which was recorded at Toronto University with two speakers (actresses aged 26 and 64 years). The dataset represents seven emotions: anger, disgust, fear, happiness, pleasant surprise, sadness, and neutrality. Both actresses were selected from the Toronto area and were university-educated with musical training. They both recorded the same number of samples, giving a total of 2,800 signals in the set. The database is balanced, and the class distribution is shown in Table 1.

4.2 Speech signal processing software

In this study, the LTAS was extracted from speech signals using PRAAT software version 6.0.39. The classifiers cited

Table 1: Class distribution in the TESS dataset

Class	Number of instances
Disgust	400
Neutral	400
Pleasure	400
Sad	400
Angry	400
Fear	400
Happy	400

earlier, implemented using the Waikato Environment for Knowledge Analysis version 3.7.9, were used for emotion recognition.

4.3 Classification algorithms

The algorithms with the best performance in our experiments, i.e. MLP, KNN, RF, SL, and SMO, will be briefly described in this section.

4.3.1 MLP

MLP is an architecture inspired by a biological neural network using two or more neurons called perceptron, which is organised in the form of layers [42], where each perceptron in a layer is connected to a perceptron in another layer.

Feedforward connections are used between the perceptron layers; the signal flows in only one direction, from the input layer to the output layer, passing through the hidden layers. Each perceptron is fully connected to each perceptron in the next layer, where connections have individual adjustable weights that simulate synapses (Figure 4).

A perceptron function could be defined by (6):

$$f = \sum_{i=1}^{n} \cdot W_i^n \cdot X_i^n + W_\Theta = W^T \cdot X. \tag{6}$$

The MLP is trained using the backpropagation algorithm, where the error is propagated from the output to inner nodes, and the weights are then adjusted using a descending gradient, in order to minimise the output error.

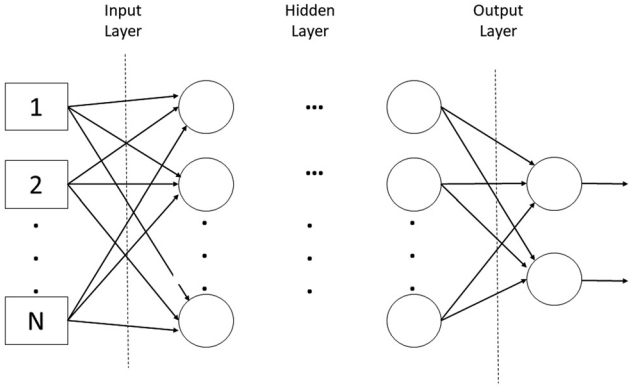

Figure 4: Structure of the MLP.

The error e in the jth neuron, where d is the desired value and y is the output value, is determined by

$$e_j(n) = d_j(n) + y(n). \tag{7}$$

A variety of neural nets have been used for emotion recognition with successful results [13,14,43]. MLP has been used in this area; many works in the literature survey their structure [15] and have highlighted their relatively high performance. The principal characteristics of MLP are (i) the use of an activation function for each neuron, (ii) a minimum of three layers (input, hidden, and output), and (iii) the learning process.

4.3.2 KNN

The KNN proposed by Cover and Hart [44] and is a non-parametric method based on similarity measures in which a distance function is applied and the input vector (unclassified instance) is assigned the class label of its nearest neighbour. The input vector is compared against each instance of the training set TS, using a measure as Euclidean distance, to search for the N nearest instances to the input vector, which label it as the majority class of the N instances. The most typically used distance measure is defined as follows:

$$d(x, x') = \sqrt{\sum_{i=1}^{N} \cdot (x_i - x_i')^2}, \tag{8}$$

where x' is the input vector, and x is each instance in the TS previously labelled with a class, both with the same N dimensionality. The set of the K minimal distances from x to x', as expressed in (9), takes into account the size of the training set (TS):

$$K_{\min}\{d(x, x')\} \quad \forall (x_j)(x_j) \in TS \tag{9}$$

The KNN classifier is a simple and older classifier that is widely used in pattern recognition. It was used in our experiments with significant results, as detailed in the experimental section.

4.3.3 Random forests

Leo Breiman was the first to provide a text-based definition of a RF [45] on his inital idea [46], as follows:

> Definition: A random forest is a classifier consisting of a collection of tree structured classifiers $h(x, \Theta_k)$, $k = 1, \ldots$ where the Θ_k are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x.

Each tree grows using a random vector Θ_k and the training set, building a classifier $h(x, \Theta_k)$. Θ_k is generated

independently from previous random vectors $\Theta_1, \ldots, \Theta_k - 1$ with the same distribution for all the trees, giving a set of tree classifiers. The final response is determined based on an evaluation of the individual responses of all the trees. The effectiveness of each tree depends on the strength of the trees in the forest and the correlations between them. The first step in creating the trees is a random selection, with replacement of features from the samples in the training set, where a different subset of data is used to develop each decision tree model. In the second step, features are randomly selected to split each node [47].

Tin Kam Ho applied an extended version of a random subspaces method [48,49], created by Breiman et al. [50]. C4.5 algorithm of Ross Quinlan [51], a version based on ID3 algorithm, was developed by himself. J48 was implemented in Weka based in C4.5.

4.3.4 SL

SL is a classifier based on the LogitBoost algorithm, where the features are selected by applying simple regression functions to LogitBoost. The J-class LogitBoost algorithm uses quasi-Newtonian steps for fitting and an additive symmetric logistic model based on the maximum likelihood [52], where the probability is expressed as follows:

$$P_{j(x)} = \frac{e^F_{j(x)}}{\sum_{k=1}^{J} \cdot e^F_{k(x)}}. \tag{10}$$

The simple logistic model is given as:

$$\text{logit}(\pi) = \ln\left(\frac{\pi}{1-\pi}\right) = \infty + X\beta. \tag{11}$$

This approach allows for the creation of simple models while preventing over-fitting of the training data [53]. The algorithm works as follows. LogitBoost initially builds a model for the root node. The iterations of the algorithm are determined by a five cross-fold validation. The algorithm is run on the training set with a maximum of 200 iterations and is used to build logistic regression models [54]. The data are divided based on C4.5 splitting criteria, and LogitBoost is used to build logistic regression models at the child nodes.

4.3.5 SMO

The SMO algorithm is an improved version of the support vector machine (SVM) algorithm for large quantities of data. The algorithm replaces all missing values, transforms nominal attributes into binary format, and normalises all attributes by default.

Overview: first, run all samples, then find a Lagrange multiplier that violates the Karush–Kuhn–Tucker (KKT) conditions (linear equality constraint), which is eligible for optimisation. Next, it chooses a second multiplier, randomly finds the value (a) from another sample, optimises the value (a), solves for the two multipliers, and updates the new optimal values in SVM. Finally, these steps are repeated until the KKT conditions are fulfilled, i.e., if the iteration is greater than M (of Vapnik), it ends, otherwise, it is repeated to all samples again. SMO is an analytical method of optimisation that solves the SVM quadratic programming (QP) problem by decomposing it into QP subproblems.

4.4 Classifier settings

The optimal settings of the principal parameters of the classifiers used in our experiments are detailed here.

The KNN was set based on the Euclidean distance, with no distance weighting, with one and three neighbours, denoted here as KNN and KNN3, respectively. The parameters for the MLP were set to 0.3 for the learning rate, 0.2 for the momentum, and 500 epochs. A single hidden layer was used in the experiments. A criterion of (classes + input size)/2 was used to determine the number of neurons in the hidden layer; for instance, using seven classes and 70 Hz for LTAS extraction gave 175 features and seven classes, where the number of neurons in the hidden layer was 91.

The SMO used multinomial logistic regression where the complexity parameter was set to one, and an epsilon value of 1.0×10^{-12} was used for the round-off error. The main parameters used for the RF classifier were as follows: bagPercentSize (percentage of the training set size) was set to 100, batchSize (number of instances to be processed) was set to 100, numIterations (number of trees in the forest) was set to 100, numExecutionSlots (number of threads used) was set to one, numFeatures (number of features chosen randomly to build the trees) was set to zero, which apply equation (12) to obtain the number of features (attributes). In experiments TotalAttributes is 175, and then eight features were used. The parameter maxDepth, which denotes the depth of the trees, was set to zero (for unlimited depth).

For the SL classifier, the batchsize was set to 100, and the parameter heuristicStop was set to 50 as the number of iterations if no new minimal error was found. The

parameter maxBoostingIterations was set to 500, and its value was directly proportional to the dataset size. The parameter weightTrim was set to zero, meaning that no trim beta was used. The NB and BN classifiers did not require parameter settings for these experiments.

$$\text{numFeatures} = \log_2(\text{TotalAttributes}) + 1. \quad (12)$$

4.5 Framework

The speech emotion signals were read using PRAAT software, and the LTAS Praat function was applied, which required the parameter of the bandwidth (BW). We used BW values of 70, 100, 200, and 300 Hz, respectively (see experiments for details), which was done using a PRAAT script. In order to start the LTAS extraction, the speech signal was split into frames and windowed, where the last was used to minimise spectral leakage. A FFT was then applied to each window. The entire signal was split using a bin width or a frequency step in order to obtain the logarithmic power spectral density from the frequencies, expressed as dB/Hz: 2×10^{-5} Pa^2 [25]. A set of LTAS values were returned for each speech signal, and the LTAS vector was then built, with labels for the emotions specified in the database. A database with LTAS values was built with a cardinality equal to the number of emotional speech signals in the TESS database; this value was 2799, as one file in the "Fear" class could not be read correctly and was ignored. The LTAS database was then processed in WEKA, using different classifiers with their default parameters. A 10-fold cross validation process was used for the classification. The entire speech emotion recognition framework is illustrated in Figure 5.

A brief description of the algorithm used in the framework is provided in the following:

(1) **Begin**
(2) **For each file in Data Set**:
  (a) Load the audio file and resample to 44 kHz.
  (b) Perform frame blocking:
    (i) Divide the audio signal into overlapping frames, reducing spectral distortion.
    (ii) Set `frame_size` = 20 ms and `frame_overlapping` = 10 ms.
  (c) Apply a Hamming window to each frame to reduce spectral leakage.
  (d) Compute the Short-Time Fourier Transform (STFT) to obtain a matrix of spectral values:
    (i) `STFT[k, n]` = FFT(`Frame[n]`), where k is the frequency bin and n is the frame index.
    (ii) Compute the magnitude spectrogram `STFT[k, n]`.
  (e) Calculate the LTAS:
    (i) `LTAS[k]` = $\frac{1}{N}\sum_{n=1}^{N}|\text{STFT[k,n]}|$, across all frames.
  (f) Generate the LTAS vector (feature vector).
(3) **End For each**
(4) With the `LTAS[k]` vector of each file, train the classifier using ten-fold cross-validation.
(5) Predict the emotion of the input audio file using the trained classifier.
(6) **End**

On the other hand, the complexity of the method will be defined by abstracting the previous algorithm as follows:

(1) Load N files and resampling, with complexity $O(N \times L)$, where:
  (a) N is the size of the data set (number of files).
  (b) L is the average frame length of the audio.
(2) Compute the STFT:
  (a) **Blocking:** Divide the signal into frames without overlap, with complexity $O(N \times L)$.
  (b) **Windowing:** Apply a window to each segment (frame), with complexity $O(N \times L)$.
  (c) **STFT computation:** Perform the STFT with complexity $O(N \times L \log(\text{frame\_size}))$, where:
  –`frame_size` is the length of the frame in milliseconds.
(3) Calculate LTAS: Averaging the spectral data of each division (frame), with complexity $O(B \times \frac{L}{\text{frame\_size}})$, where:
  (a) B is the bin width of frequency.

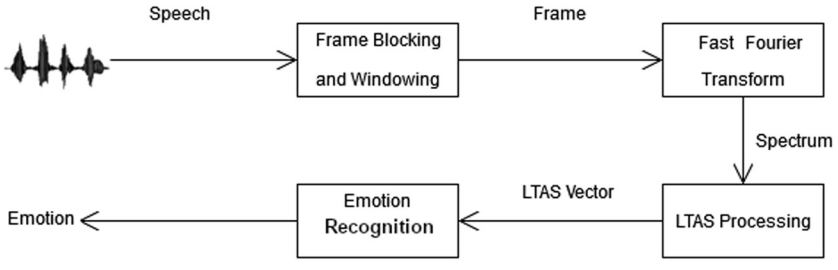

Figure 5: Proposed speech emotion recognition framework.

(b) The complexity of N files is $O(N \times B \times \frac{L}{\text{frame\_size}})$.

(4) Generate the LTAS vector for N files using B bins, with complexity $O(N \times B)$.

(5) Perform tenfold cross-validation: assuming C as the complexity of training the classifier, the overall complexity is $O(10 \times C) = O(C)$.

(6) Predict emotion: the complexity depends on the classifier used.

Summarising the computational complexity analysis in the process:

(1) **SFFT computing:** The computational complexity is given by

$$O(N \times L \log(\text{frame\_size})),$$

where

– N is the number of files,
– L is the average frame length of the audio,
– `frame_size` is the length of each frame.

(2) **LTAS averaging:** The computational complexity is as follows:

$$O\left(N \times B \times \frac{L}{\text{frame\_size}}\right),$$

where

– B is the bin width of the frequency,
– N is the number of files,
– L is the average frame length of the audio,
– `frame_size` is the length of each frame.

The representative complexity of the method is, using N files, as follows:

$$O\left(N \cdot (L \cdot \log(\text{frame\_size}) + B \cdot \frac{L}{(\text{framesize})})\right).$$

The method takes advantage of the low computational load of LTAS processing, unlike other schemes such as MFCC, which require additional steps, such as applying the Mel filter bank, logarithmic transformation, and discrete cosine transformation.

5 Results and discussion

LTAS processing gives a reduced abstract of the frequency spectrum, where the number of values per second is significantly lowered. In the last module, a classifier is applied for classification, which in our case involves determining which of the seven emotions fits the speech signal. Typically, in schemes reported in the literature, the audio signals are passed through a pre-processing phase, where a pre-emphasis filter and noise reduction are typically applied. In a data mining approach, many filters could be applied to the data, such as normalisation, attribute selection and resampling filters. In this work, neither audio signal filters nor data mining filters were applied to the samples from the TESS database.

5.1 Feature extraction

The feature vectors were created by extracting the LTAS from the audio signals. LTAS extraction requires a BW frequency, as this defines the number of values (descriptors) used to characterise the audio signal. BW values of 70, 100, 200, and 300 Hz were used in this case, giving 175, 122, 62, and 40 LTAS values, respectively. In our experiments, the larger the value used for the LTAS, the better the performance.

5.2 Comparative analysis

Algorithms based on trees, functions, SVM, logistic regression, Bayes, KNN, and NNs were tested, and a tenfold cross-validation method was applied. Table 2 presents the performance of each classifier, using a defined number of descriptors (values) as indicated in the header. In these experiments, the MLP gave the highest performance for each BW frequency in the sampling process. Increasing the LTAS values improved the performance of the SMO and SL classifiers, particularly at BW values of 70 and 100 Hz, where SMO performed just slightly below MLP.

The KNN classifier was robust over different attribute sizes (first row), giving similar performance. The RF among the set of trees algorithms, is outstanding with an accuracy of between 94 and 98%. The Bayesian algorithms gave the lowest performance on emotion classifications using LTAS. Although a set of other algorithms were tested, only the best performing of each type are presented here. For each classifier, the average accuracy for the different BWs was computed to determine the overall performance across the range of BWs. The last column in Table 2 contains the average accuracy, and it can be seen that MLP was the best classifier for emotion recognition using the TESS database, followed by KNN (with a difference of approximately 2%), and then the RF, SMO, and SL classifiers of Figure 6.

The classifiers based on Bayes gave the lowest performance. In the discussion section, we describe state-of-the-art works where relevant results were obtained using variants of NN, SVM, and KNN. The following works involved experiments with the TESS database, where a subset of features as MFCC, filter bank energies (FBEs), energies,

Table 2: Performance of algorithms using four bandwidths for LTAS extraction

Clf.	175 70 Hz	122 100 Hz	62 200 Hz	40 300 Hz	Avg. Acc.
MLP	99.24	99.03	98.03	96.28	98.14
KNN	96.99	96.64	96.17	95.31	96.27
RF	97.89	94.03	96.17	95.14	95.80
SMO	98.42	97.46	94.67	91.89	95.61
SL	98.35	97.53	93.92	91.03	95.20
Logistic	92.56	91.74	92.10	90.06	91.61
J48	90.49	87.92	86.88	84.03	87.33
RT	81.56	81.27	81.63	79.81	81.06
BN	75.63	74.14	73.24	71.84	73.71
NB	52.69	48.41	46.12	44.69	47.97

Source: Authors.

Table 3: Performance comparison with state-of-the-art works

Classifier	Main speech codif.	Feats	Values used	Classes	Classif (%)
SVM [55]	MFCC, LPC, F_0, Intensity, Tempo,	9	140	7	64.20
SVM [56]	MFCC	1	12*	4	97.00
GRU [13]	FBE	1	26*	5	95.82
KNN [19]	MFCC and FBE	13	637+	6	95.00
KNN [39]	MFCC	1	13*	7	84.00
SVM [39]	MFCC and Chroma	11	25*	7	90.00
DNN [13]	FBE	1	26*		89.96
MLP	LTAS	1	175+	7	99.24
MLP	LTAS	1	122+	7	99.03
SMO	LTAS	1	175+	7	98.42
SL	LTAS	1	175+	7	98.35
MLP	LTAS	1	62+	7	98.05
RF	LTAS	1	175+	7	97.89
KNN	LTAS	1	175+	7	96.99

Notes: *Values used for each frame, where each frame long commonly is 20 ms. +Values used for entire signal.
Source: Authors.

zero-crossing rates, and chroma coefficients were used. A comparison of performance between these works and our framework is presented in Table 3.

Experimental results with a subset (1,369 samples) of the TESS database were reported by Praseetha and Sagil [13], who used 26 FBE coefficients per frame. The total number of values used per signal was not specified, but long frames are typically 20 ms (overlapped by 10 ms), with 2,600 values per second. The best performance was 95.82%, achieved using a GRU, an improved version of an RNN. In our experiments, the best performance was obtained using a common kind of NN called an MLP.

Experiments were also reported by Ugu Bastug [19], who used coefficients such as delta values, acceleration, mean, standard deviation, maximum and minimum of MFCC, and the median, standard deviation, and acceleration of the FBEs. In total, a matrix of values with 49 rows and 13 columns was used, although it was not specified whether this matrix represented the complete audio signal or a frame. Assuming that the matrix represents the complete signal (2 s long), this would give 637 values. In summary,

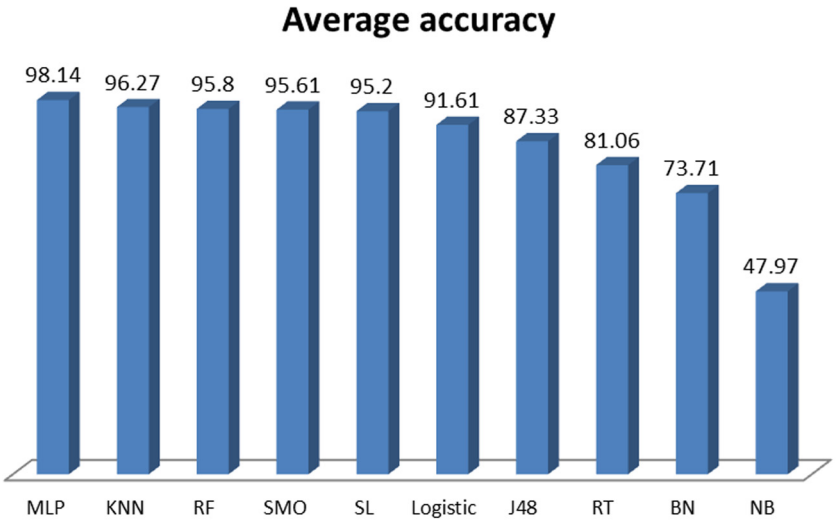

Figure 6: Averaged accuracy for each classifier over values of 175, 122, 62, and 40 for the LTAS per signal.

signal processing for the numeric characterisation was based on MFCC and FBE codification, in addition to the statistical measures cited earlier. The KNN classifier was used in these experiments similarly to our experiments, and one of the best results, reported in Table 3, was obtained using KNN.

Experiments with the TESS database, SVM classifier, and MFCC as speech features for speech emotion recognition were carried out by Zafar Iqbal and Farooq Siddiqui [56]. A subset of TESS samples containing four emotions (anger, happiness, disgust, and pleasant surprise) were used for prediction purpose, and 50 and 6 samples were used for each emotion, respectively, for training and testing. These authors argued that with a subset, it was possible to predict the behaviour of the entire database. The MFCC extraction process used rectangular segments of 10 ms (100 per second), and the number of coefficients was 12, giving an estimate of 1,200 values of MFCC per second.

This work was used as a comparison with the results of our method, as the same database was tested (TESS) with different features (our method uses LTAS). When used for emotion recognition, MFCC and LTAS features have the following differences: (i) a method based on LTAS uses 62 to 175 values per signal, as shown in Table 3 (the number of values varies as a function of the BW frequency in LTAS extraction process), whereas the method based on MFCC uses 1,200 values per s; (ii) the accuracy of correct emotion classification in our method ranges from 98.05% using 62 values per signal to 99.24% using 175 values per signal, when the MLP classifier is used, compared with an accuracy of 97% with the method of Zafar Iqbal and Farooq Siddiqui method, which is based on SVM. Our method achieves an accuracy of 98.42% using SMO, a kind of SVM.

Sundarprasad [39] experimented with the entire TESS database, where 34 values of the speech signal were extracted after the application of audio analysis. Some of the features used were MFCC, energy, measures of spectral coefficients, zero crossing rate, and chroma coefficients, with a total of 11 features. Principal components analysis (PCA) was applied to reduce the dimensionality from 34 to 25. The classifier used was SVM, and in a similar way to our experiments, the third best performance was obtained using SMO, an improvement on the SVM classifier, which achieved an accuracy of 98.42% correct classification, using only 175 values for the entire signal.

The work presented in [55] involved the Spanish databases EmoSpanishDB and EmoMatchSpanishDB. MFCCs were extracted from the audio files, with 13 per frame (with frames 20 ms long). The mean, first and second derivative, spectral centroid, and spectral contrast were computed from the MFCC values, and lineal spectral coding (LPC) values, intensity, and fundamental frequency (F0) were used in this method. The best results were obtained from SVM, with an accuracy of 56.04% for recognising the emotions of disgust, happiness, fear, sadness, surprise, anger, and neutrality.

Features are generally high dimensionality, as can be observed from Table 3. There have been research studies in this area [1], where architectures based on DNNs were developed in order to extract, reduce, and homogenise features. Features such as IS10, MFCC, and eGemspd were used and tested using the KNN, logistic regression, RF and SVM classifiers. In our experiments, we also found that these algorithms were the best for the task of emotion recognition. The main difference lay in the order of their performance, as Jiang [1] reported that the SVM achieved the best performance, whereas the MLP was superior in our experiments.

In summary, the experiments carried out in this work yielded similarities with the classifiers applied in state-of-the-art schemes on the TESS database was used. We considered the classifiers NN (GRU), SVM, and KNN. We also compared different speech codifications (MFCC, FBE, chroma, and LTAS) using the TESS database. In Table 3, which shows the results for our method based on LTAS, the number of values extracted for the entire signal is denoted with a + sign, and the number of values used for the other methods for each frame of 20 ms is denoted by a * sign. For the entire signal, the number of values used for emotion recognition, in the state-of-art methods, is higher.

6 Conclusions

A new framework based on LTAS was tested for speech emotion recognition using the TESS database, and was found to give similar or higher performance than methods/frameworks based on traditional coding schemes (MFCC, LPC, FBE). Our results show that the use of LTAS allows us to significantly reduce the number of values and features per signal, thereby achieving a low computational load and fast response time. We also conducted comparative experiments among classifiers such as NN, KNN, logistic, Bayesian trees, and SVM. Experiments performed with LTAS as a speech coding scheme and MLP as a classifier showed a high level of effectiveness, with an accuracy of 99.32% in terms of correct emotion recognition. SMO and SL achieved an accuracy of 98%, in the best case and with 175 coefficients, while KNN and RF achieved accuracies of 96.99 and 97.85%, respectively, using 175 values of LTAS. The performance of our model was higher than those reported as the state-of-art. Classifiers for speech emotion recognition such as KNN, NN,

and SVM in this work were the same classifiers reported in the state-of-art works, and provided high speech emotion recognition accuracy.

The method was tested on short spoken sentences recorded in a controlled environment without noise. It must be improved for robustness to perform in noisy or real-world environments. This study tested whether LTAS is useful for emotion recognition, as this task does not require highly detailed spectral information. However, for other tasks, such as speech recognition, LTAS may not be suitable, and other schemes such as MFCC are more efficient.

Acknowledgement: We thank the Universidad Autónoma del Estado de Quintana Roo and Universidad del Istmo for the use of its facilities and equipment and the Red Iberoamericana de Academias de Investigación A. C. for sponsorship.

Funding information: We thank the Universidad Autónoma del Estado de Quintana Roo and Universidad del Istmo for the use of its facilities and equipment and the Red Iberoamericana de Academias de Investigación A. C. for sponsorship.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and approved its submission.

Conflict of interest: The authors state that there is no conflict of interest.

Data availability statement: The data described in this article are available in the University of Toronto Dataverse at https://borealisdata.ca/dataset.xhtml?persistentId=doi:10.5683/SP2/E8H2MF.

References

[1] W. Jiang, Z. Wang, J. S. Jin, X. Han, and C. Li, "Speech emotion recognition with heterogeneous feature unification of deep neural network," *Sensors* vol. 19, no. 12, p. 2730, 2019.

[2] R. W. Picard, *Affective computing*, Cambridge, MA, USA: MIT Press, 1997.

[3] C. L. P. E. Greco Alberto Valenza Gaetano, "Arousal and valence recognition of affective sounds based on electrodermal activity," *IEEE Sensors Journal*, vol. 17, pp. 716–725, 2017.

[4] R. Beale and C. Peter, *Affect and emotion in human-computer interaction*, Springer, 2008. https://doi.org/10.1007/978-3-540-85099-1.

[5] D. Hill, *Emotionomics: Winning Hearts and Minds*, Beavers Pond Press, 2007, https://books.google.com.mx/books?id=RpyYGQAACAAJ.

[6] R. Picard, E. Vyzas, and J. Healey, "Toward machine emotional intelligence: analysis of affective physiological state," *IEEE Trans. Pattern Anal. Machine Intel.*, vol. 23, no. 10, pp. 1175–1191, 2001.

[7] E. Hudlicka, "To feel or not to feel: The role of affect in human-computer interaction," *Int. J. Human-Comput. Stud.*, vol. 59, no. 1, pp. 1–32, 2003, https://www.sciencedirect.com/science/article/pii/S1071581903000478.

[8] F. Chiara, F. Adolfo, P. Rocco, P. David, C. Daniela, C. Irene, et al., "Automated affective computing based on bio-signals analysis and deep learning approach," *Sensors*, vol. 22, no. 5, p. 1789, 2022. https://www.mdpi.com/1424-8220/22/5/1789.

[9] A. Landowska, "Affective computing and affective learning - methods, tools and prospects," *EduAkcja. Magazyn Edukacji Elektronicznej*, vol. 1, no. 5, pp. 16–31, 2013.

[10] C. Irene, P. Rocco, C. Adolfo, L. Malva Pasquale, M. Daniela, M. Roberta, et al., "Age-related differences in the perception of covid-19 emergency during the Italian outbreak," *Aging Mental Health*, vol. 25, p. 10, Dec 2020.

[11] R. Reisenzein, "Wundtas three-dimensional theory of emotion," in *Structuralist Knowledge Representation*, Brill, 2000, pp. 219–250. https://doi.org/10.1163/9789004457805_012.

[12] L. Kerkeni, Y. Serrestou, K. Raoof, M. Mbarki, M. Mahjoub, and C. Clender, "Automatic speech emotion recognition using an optimal combination of features based on EMD-TKEO," *Speech Commun.*, vol. 114, no. 1, pp. 22–35, Sep 2019.

[13] V. Praseetha and S. Vadivel, "Deep learning models for speech emotion recognition," *J. Comput. Sci.*, vol. 14, no. 11, pp. 1577–1587, 2018.

[14] A. Shaw, R. Kumar, and S. Saxena, "Emotion recognition and classification in speech using artificial neural networks," *Int. J. Comput. Appl.*, vol. 145, no. 8, pp. 5–9, 2016.

[15] P. Hemanta, M. Mihir, and C. Mahesh, "Use of different features for emotion recognition using mlp network," in *Advances in Intelligent Systems and Computing, I. Sethi, Ed*. Springer India: Springer, New Delhi, 2015.

[16] S. Karimi and M. H. Sedaaghi, "Best features for emotional speech classification in the presence of babble noise," in *20th Iranian Conferance on Electrical Engineering. Tehran*, IEEE, Iran, 15–17 May 2012, pp. 1047–1051.

[17] S. Emerich and E. Lupu, "Improving speech emotion recognition using frequency and time domain acoustic features," in *Proceedings of SPAMEC*. EURASIP, Cluj-Napoca, Romania, 26–28 August 2011, pp. 85–88.

[18] S. K. Jagtap, V. V. Nanavare, Recognition of human emotions from speech processing, *International Conference on Advances in Computing, Communication and Control*, vol. 49, pp. 24–32, 2015.

[19] U. Bastug, On classification and recognition of emotions by investigating equivalent speech and textual data, Master's thesis, University of Westminster, 2016.

[20] V. Pappu and P. Pardalos, *High dimensional data classification*, Springer, Dec 2013, p. 34. https://doi.org/10.1007/978-1-4939-0742-7_8.

[21] A. Othman, T. Hasan, and S. Hasoon, "Impact of dimensionality reduction on the accuracy of data classification," in *2020 3rd International Conference on Engineering Technology and its Applications (IICETA)*, Najaf, Iraq, Sep 2020, pp. 128–133.

[22] M. R. Hassan, B. Nath, and M. A. Bhuiyan, "Bengali phoneme recognition: a new approach," in *Proc. 6th International Conference on Computer and Information Technology (ICCIT03)*, 2003.

[23] C. Hao, M. Xin, and Y. Xu, "A study of speech feature extraction based on manifold learning," *J. Phys. Confer. Ser.*, vol. 1187, p. 052021, April 2019.
[24] P. Gambhir, A. Dev, P. Bansal, and D. K. Sharma, "End-to-end multi-modal low-resourced speech keywords recognition using sequential conv2d nets," *ACM Trans. Asian Low-Resource Language Inform. Process*, vol. 23, no. 1, pp. 1–21, 2024.
[25] P. Boersma and D. Weenink, *Praat: Doing phonetics by computer*, 2010. http://www.praat.org/.
[26] H. Luis, "Speech segmentation with text independence, using time domain features," *Congreso Internacional de Tecnologiiias Inteligentes y de Información*, vol. 38, pp. 129–139, Oct 2008.
[27] S. S. Stevens and J. Volkmann, "The relation of pitch to frequency: A revised scale," *Amer. J. Psychol.*, vol. 53, no. 3, pp. 329–353, 1940. http://www.jstor.org/stable/1417526.
[28] H. Luis, H. Jose, and C. Julio, "Speech segmentation algorithm based on fuzzy memberships," *Int. J. Comput. Sci. Inform. Security*, vol. 8, no. 1, pp. 229–233, April 2010.
[29] H.-S. Kim, *Linear predictive coding is all-pole resonance modeling,* Center for Computer Research of Music and Acoustic, Stanford University, Stanford, 2014. https://api.semanticscholar.org/CorpusID:34456318.
[30] N. Dave, "Feature extraction methods LPC, PLP and MFCC in speech recognition," *Int. J. Adv. Res. Eng. Tech.*(ISSN 2320-6802), vol. 1, July 2013.
[31] J. Tukey, D. Brillinger, and D. Cox, *The Collected Works of John W. Tukey: Time series: 1965–1984*, ser. Wadsworth statistics/probability series. Wadsworth Advanced Books and Software, 1994. https://books.google.com.mx/books?id=ef7uAAAAMAAJ.
[32] R. Cohen and Y. Lavner, *Infant cry analysis and detection*, IEEE, Dec 2012, pp. 1–5. https://doi.org/10.1109/EEEI.2012.6376996.
[33] MICAI'05:*Proceedings of the 4th Mexican International Conference on Advances in Artificial Intelligence*. Springer-Verlag, Berlin, Heidelberg, 2005.
[34] J. R. Deller, J. G. Proakis, and J. H. L. Hansen, *"Discrete-time processing of speech signals,"* Prentice Hall PTR, 1993. https://dl.acm.org/doi/abs/10.5555/562892.
[35] T. Kinnunen, V. Hautamäki, and P. Fränti, "On the use of long-term average spectrum in automatic speaker recognition", in *5th International Symposium on Chinese Spoken Language Processing (ISCSLP'06*, 2006, pp. 559–567. https://www.isca-archive.org/iscslp_2006/kinnunen06b_iscslp.html.
[36] S. Cukier-Blaj, Z. Camargo, and S. Madureira, *Long-term average spectrum loudness variation in speakers with asthma, paradoxical vocal fold motion and without breathing problems,* Speech Prosody, Jan 2008, pp. 41–44. https://doi.org/10.21437/SpeechProsody.2008-9.
[37] P. Rose, *Forensic speaker identification*, CRC Press, London, 2002. https://doi.org/10.1201/9780203166369.
[38] J. Pittam, *Voice in social interaction: An interdisciplinary approach*, Sage Publications, London, 1994.
[39] N. Sundarprasad, Speech emotion detection using machine learning techniques, Master's thesis, San Jos State University, United States, May 2018.
[40] D. Pravena and D. Govind, "Development of simulated emotion speech database for excitation source analysis," *Int. J. Speech Tech.*, vol. 20, no. 2, pp. 327–338, 2017.
[41] M. K. Pichora-Fuller and K. Dupuis, *Toronto emotional speech set (TESS)*, 2020, doi: 10.5683/SP2/E8H2MF.
[42] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychol. Rev.*, vol. 65, no. 6, pp. 386–408, 1958.
[43] S. Latif, R. Rana, S. Younis, J. Qadir, and J. Epps, "Transfer learning for improving speech emotion classification accuracy," *Interspeech 2018*, Sep 2018, pp. 257–261.
[44] T. Cover and P. Hart, "Nearest neighbor pattern classification," *Information Theory*, IEEE Transactions on, IEEE, vol. 13, pp. 21–27, 1967. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1053964.
[45] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, p. 5–32, Oct. 2001. doi: 10.1023/A:1010933404324.
[46] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, p. 123–140, Aug. 1996. doi: 10.1023/A:1018054314350.
[47] N. Horning, "Random forests: An algorithm for image classification and generation of continuous fields data sets," in *Proceedings of the International Conference on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences*, Osaka, 2010, pp. 9–11. https://api.semanticscholar.org/CorpusID:84180341.
[48] T. K. Ho, "Random decision forests," in *Proceedings of the Third International Conference on Document Analysis and Recognition, ser. ICDAR '95*, IEEE Computer Society, USA, 1995, p. 278.
[49] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, p. 832–844, Aug. 1998. doi: 10.1109/34.709601.
[50] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees.* Wadsworth and Brooks, Monterey, CA, 1984.
[51] S. L. Salzberg, "C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993," *Machine Learning*, vol. 16, no. 3, pp. 235–240, Sep 1994, doi: 10.1007/BF00993309.
[52] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *The Annals of Statistics*, vol. 28, pp. 337–407, April 2000.
[53] Y. Sun and P. B. Kantor, *Automatic assessment of non-topical properties of text by machine learning methods,* ACM, 2005. https://dl.acm.org/doi/10.5555/1145201.
[54] M. Sumner, E. Frank, and M. Hall, "Speeding up logistic model tree induction," in *Proceedings of the 9th European Conference on European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ser. ECMLPKDD'05.* Springer-Verlag, Berlin, Heidelberg, 2005, pp. 675–683, doi: 10.1007/11564126_72.
[55] E. García-Cuesta, A. B. Salvador, and D. G. Pãez, "Emomatchspanishdb: study of speech emotion recognition machine learning models in a new spanish elicited database," *Multimedia Tools Appl.*, vol. 83, no. 5, pp. 13093–13112, 2024.
[56] M. Z. Iqbal, "MfCC and machine learning based speech emotion recognition over TESS and IEMOCAP datasets," *Found. Univ. J. Eng. Appl. Sci.*, vol. 1, no. 2, pp. 25–30, 2020.