applied sciences

MDPI

Article

An Optimal Feature Parameter Set Based on Gated Recurrent Unit Recurrent Neural Networks for Speech Segment Detection

Özlem Batur Dinler [1,2,*] and Nizamettin Aydin [2]

[1] Computer Engineering, Siirt University, Siirt 56100, Turkey
[2] Computer Engineering, Yildiz Technical University, Istanbul 34220, Turkey; naydin@yildiz.edu.tr
* Correspondence: o.b.dinler@siirt.edu.tr

Received: 5 January 2020; Accepted: 9 February 2020; Published: 13 February 2020

Abstract: Speech segment detection based on gated recurrent unit (GRU) recurrent neural networks for the Kurdish language was investigated in the present study. The novelties of the current research are the utilization of a GRU in Kurdish speech segment detection, creation of a unique database from the Kurdish language, and optimization of processing parameters for Kurdish speech segmentation. This study is the first attempt to find the optimal feature parameters of the model and to form a large Kurdish vocabulary dataset for a speech segment detection based on consonant, vowel, and silence (C/V/S) discrimination. For this purpose, four window sizes and three window types with three hybrid feature vector techniques were used to describe the phoneme boundaries. Identification of the phoneme boundaries using a GRU recurrent neural network was performed with six different classification algorithms for the C/V/S discrimination. We have demonstrated that the GRU model has achieved outstanding speech segmentation performance for characterizing Kurdish acoustic signals. The experimental findings of the present study show the significance of the segment detection of speech signals by effectively utilizing hybrid features, window sizes, window types, and classification models for Kurdish speech.

Keywords: database; deep learning; consonant/vowel/silence; segmentation; speech segment detection

1. Introduction

Speech is a sign that contains a lot of personal information. Together with the developing technology, a wide variety of applications are developed using the information obtained from this speech signal. Segmentation represents a procedure of breaking down a speech signal into smaller acoustic units. It is possible to define speech segmentation as the procedure of finding limits in a natural spoken language between words, syllables, or phonemes [1]. In this study, Kurdish phoneme segment detection is investigated. Speech segment detection is one of the most commonly used technologies for recognizing spoken words and expressions and converting them into a format that can be understood by machines, especially computers.

A phoneme, which is characterized by some distinct pertinent properties that distinguish it from other phonemes of the language, represents a language sound element. Phonemes are classified either as vowels or consonants. Sounds that are obtained by oscillating the vibrations acquired with the vocal cords in the sound path are called vowels. Consonants are the sounds produced by obstructions created to the flowing air in any position of the sound path.

In consonants, vocal cord vibrations are not important. When vocal sounds obtained with the vibration of the vocal cords are examined in the time domain, it is observed that they have a periodic structure, while non-vocal sounds have a non-periodic structure. All vowels and some consonants

are vocal. This makes it difficult to distinguish between vowel phonemes and consonant phonemes [2]. It is possible to categorize words into various types in accordance with the position of the occurrences of vowels and consonants in them. The placement of vowels and consonants in a word of any language is very significant for identifying various types of words [3]. The acoustic phoneme constitutes the basis of a wide variety of the existing speech processing systems, such as medium to large vocabulary speech recognition, speaker recognition systems, and language identification systems [4].

The accuracy of a speech processing system highly depends on the spoken language phoneme set and has more challenges due to alterations in the pronunciation of words due to dialects, the speaker's age and gender, and neighboring words. Furthermore, certain difficulties come to the forefront during the creation of a voice database. In addition, the voice segment detection process becomes more difficult due to factors including the toning effect and syllable stress. In this respect, the gated recurrent unit (GRU), which is a special recurrent neural network (RNN) model, was used to deal with these difficulties for the efficient speech segment detection. This model is now widely used in various speech processing tasks. In Ravanelli et al. [5], GRUs were reviewed and a compact single-gate model replacing a hyperbolic tangent with rectified linear unit activations having a simplified architecture was suggested for the purpose of automatic speech recognition (ASR). The long short-term memory (LSTM) and GRU were assessed for the purpose of comparing the performances they exhibited upon a reduced Technology Entertainment Design - Laboratoire Informatique de l'Université du Maine (TED-LIUM) speech data set [6]. In Cernak and Tong [7], a solution was suggested with the aim of training a phone attribute detector without phone alignment by utilizing end-to-end phone attribute modeling on the basis of the connectionist temporal classification. In Zheng et al. [8], general emotion features were produced in speech signals from various angles, and an ensemble learning model was employed for the purpose of carrying out emotion recognition tasks. The design of the expert roles of speech emotion recognition was performed by utilizing convolutional neural networks (CNNs) and GRU. In Chen et al. [9], a practical approach with three steps was introduced for singing voice detection on the basis of a GRU.

Recently, speech segment detection has been addressed in a deep neural network (DNN), CNN, and RNN models [10–12]. Franke et al. [11] examined the automatic detection of phoneme boundaries in audio recordings using deep bidirectional LSTMs. The first experiments operated on Texas Instruments Massachusetts Institute of Technology (TIMIT) and BUCKEYE datasets containing an American English language speech corpus and then a Basaa dataset containing a Bantu language speech corpus. An F1-score of 77 with a tolerance of 20 ms was achieved with the above-mentioned method. In Wang et al. [12], it is reported that GRU forget gate activations in trained recurrent acoustic neural networks correlate very well with phoneme boundaries in speech activation, which ensures the validation of numerous approaches to speech recognition and other sequence modeling problems, including recurrent networks. This study uses a TIMIT corpus. The GRU can take temporal context into account. Therefore, it should exhibit a better performance in comparison with conventional machine learning techniques [13]. Alternatively, LSTM is a slightly more complex structure variation of the GRU [14]. The GRU ensures the control of the information flow, similar to the LSTM unit, but without a need to utilize a memory unit. It only provides the exposure of the complete hidden content without any control. The GRU has a simpler structure compared to standard LSTM models, and its popularity is gradually increasing. Therefore, it is preferable for this study. Lee et al. [15] suggested phoneme segmentation by utilizing cross-entropy loss with connectionist temporal classification loss in deep speech architecture for the purpose of performing speech synthesis. In order to assess the suggested method, female Korean speech found in the Speech Information Technology & Industry Promotion Center (SITEC) speech database was utilized. The suggested method ensured the improvement in the quality of phoneme segmentation since it could provide the model with a higher number of non-blank classes or force the setting of a blank class to a non-blank class. The experimental findings demonstrated that a decrease of more than 20% occurred in the boundary error. In Graves and Schmidhuber [16], a comparison of a

bidirectional LSTM with other neural network architectures for the purpose of framewise phoneme classification on the TIMIT speech database was made. In the past, speech segment detection has been addressed in different conventional models. In Weinstein et al. [17], the development of a system to conduct the acoustic–phonetic analysis of continuous speech was performed with the aim of serving as a part of an automatic speech understanding system. In Leung et al. [18], a segmental approach was employed for the purpose of phonetic recognition. Multi-layer perceptrons were utilized to detect and classify phonemes for a segmental framework. In Ali et al. [19], the acoustic and phonetic properties of the American English stop consonants were examined. In Natarajan and Jothilakshmi [20], the segmentation of the continuous speech into smaller speech units was performed, and every unit was classified as a consonant or vowel by utilizing the formant frequencies and support vector machines (SVMs). It is reported that the Gaussian kernel yielded higher accuracy in comparison with the other kernels utilized. In Ades [21], differences between vowels and consonants and between speech and nonspeech with regard to the context range in which they are set were explained. Identification, discrimination, and delayed recognition tests, in which the discrimination of vowels and consonants was performed at various levels, were explained regarding the higher range of the vowel series over the consonant series and the tasks' memory demands. In Ooyen et al. [22], two tests with vowels and consonants as the targets of phoneme detection in real words were performed. In the first test, the comparison of two comparatively distinct vowels with two confusable stop consonants was performed. In the second test, the comparison of two comparatively distinct vowels with the semivowels corresponding to them was carried out. The two tests demonstrated that the detection of English vowels was difficult in comparison with stop consonants but easy in comparison with semivowels. A novel method to segment phonemes by utilizing multilayer perceptron (MLP) was suggested in Suh and Lee [23]. The suggested segmenter's structure includes three parts, i.e., a preprocessor, an MLP-based phoneme segmenter, and a postprocessor. The pre-processor utilized feature parameters for every speech frame in accordance with the acoustic phonetic information. The MLP-based phoneme segmenter was utilized for the purpose of learning the ability to determine the boundaries in question and segment continuous speech into corresponding phonemes. In post-processing, the positions of the phoneme boundaries were decided by utilizing the MLP output. The best performance of phoneme segmentation was found to be approximately 87% with an accuracy of 15 ms.

In recent years, more researchers have effectively combined CNN and RNN models [24]. These models were applied to different speech processing tasks to improve the accuracy [25–29]. In general, it is beneficial to use CNN and RNN methods together. Therefore, in our work we used this model.

This paper presents an extensive and first study for finding optimal feature parameter sets by using the GRU for Kurdish speech segment detection. Also, it is the first time speech detection has been developed for the Kurdish language. The effects of varying hybrid features, window type, window size, and different classification methods were investigated for Kurdish speech segment detection. The identified optimal parameter features were analyzed using a novel Kurdish speech dataset collected from a television broadcast.

Kurdish represents the language that is spoken by an estimated seventeen million speakers in Turkey, Armenia, Syria, Azerbaijan, Iran, and Iraq. It is the most widely spoken language in Middle East after Arabic, Turkish, and Persian. Therefore, the speech segment detection study to be conducted on Kurdish is important.

2. Materials and Methods

Kurdish sound samples were collected from both males and females. Prior to phoneme segmentation, the continuous speech signal was segmented into words.

The phoneme segment detection of the spoken language has mainly focused on utilizing short-time energy and short-time zero crossing rate (ZCR) features. Since errors cause a low accuracy in speech segment detection, an effective algorithm should be used for the segment detection of the Kurdish speech at different speaking speeds. The mel frequency cepstral coefficient

(MFCC) is known to mimic human hearing system [30]. MFCC coefficients are orthogonal to each other and the mel filter bank also models the perceptions of the human ear system, and it yields better results in comparison with other feature extraction methods. Therefore, we used a hybrid feature vectors of the voice samples. Evaluation and comparison of the system performance were performed on the basis of energy, ZCR, and MFCC along with its first- and second-order derivatives. Thus, we used three types of hybrid feature vectors: energy, ZCR, and MFCC (EZMFCC); energy, ZCR, and delta-MFCC (EZDMFCC); and energy, ZCR, and delta-delta-MFCC (EZDDMFCC). Hybrid feature vectors were extracted using Hamming, Hanning, and rectangular windowing with 20, 25, 30, and 35 ms window sizes of the speech segment to see the effect of the different window types and window sizes with GRU training on the classification performance of the classifiers. For classification, convolutional neural network (CNN), multilayer perceptron (MLP), and the standard classifiers naive Bayes, random forest (RF), support vector machine (SVM), and k-nearest neighbors (k-NN) were used. Here, the details regarding the Kurdish speech corpus preparation and implementation of the segment detection architecture are explained. For this operation, the PRAAT 6.0.49 tool [31], MATLAB R2018a (MathWorks, Natick, MA 01760-2098, USA), Keras Library [32], Weka 3.9.3 [33], and WekaDeeplearning4j [34] are used.

2.1. Kurdish Phoneme Set and Its Properties

Kurdish is part of the Western Iranian group of the Indo-Iranian branch of the Indo-European language family. It is considerably similar to European languages, such as French, English, Russian, and German. Nevertheless, Kurdish is most similar to Persian among the European languages. However, both languages have a completely unique vocabulary, morphological structures, phonetic properties, and grammatical structures [35]. Based on Latin letters, Kurdish consists of thirty-one phonemes, including twenty-three consonants and eight vowels. Five vowels of Kurdish are accepted as long vowels (/a/, /ê/, /î/, /û/, /o/), and the other three are accepted as short vowels (/e/, /i/, /u/). Despite the fact that they are generally named as long and short vowels, vowels are presently differentiated according to the position of articulation. There are a total of twenty-three consonants, including six labial consonants (/b/, /f/, /m/, /p/, /v/, /w/), seven fronto-palatal and dental consonants (/d/, /l/, /n/, /r/, /s/, /t/), five palatal consonants (/c/, /ç/, /j/, /ş/, /y/), and five anterior palatal and laryngeal consonants (/g/, /k/, /h/, /q/, /x/). In this paper, we detected phonemes consisting of the silent, eight vowels, and twenty-three consonants. The phonemes of Kurdish are presented in Table 1, in which Kurdish letters, International Phonetic Alphabet (IPA) notations, and phonetic descriptions are given [35–37].

2.2. Speech Dataset

In this study, a novel Kurdish dataset was created and used. The speech corpus was a collection of speech recordings from Turkish Radio and the Television Corporation (TRT) Nûçe news channel. For this study, the speech corpus was selected from different Kurdish sources, such as education news, culture news, art news, economy news, health news, and political news. The collected Kurdish voice samples of sentences and clauses of different lengths were taken from four male and three female speakers in the age group of 20–45 years. The speech sentences contained CV (consonant–vowel), VC (vowel–consonant), CVC (consonant–vowel–consonant), CCV (consonant–consonant–vowel), CVCC (consonant–vowel–consonant–consonant) and CCVCC (consonant–consonant–vowel–consonant–consonant) as different types of words. The recordings were performed in a quiet setting using a high-quality (low noise) desktop microphone at a sampling frequency of 44,100 Hz with 16-bit quantization and a mono-channel. Speech data were stored in the "wav" format. The whole speech corpus was randomly split into two parts: 66% for training, and the remaining 33% for testing the system.

The phoneme segment detection method was implemented on the continuous Kurdish speech containing almost 6819 phonemes in total. Table 2 and Table 3 present the dataset of phonemes from male and female speakers, respectively.

Table 1. Phonemes of the Kurdish language [35–37].

Kurdish Letter	IPA	Phonetic Description
a	[ɑ]	Open, back, unrounded vowel
b	[b]	Bilabial voiced stop
c	[dʒ]	Palatal voiced affricate
c	[tʃ]	Palatal voiceless unaspirated affricate
	[tʃh]	Palatal voiceless aspirated affricate
d	[d]	Alveo-dental voiced stop
e	[ae]	Near-open, front, unrounded vowel
	[ɛ]	Open-mid, front, unrounded vowel
ê	[e]	Close-mid, front, unrounded vowel
	[ə]	Mid-central unrounded vowel
f	[f]	Labiodental voiceless fricative
g	[g]	Velar voiced stop
h	[h]	Glottal fricative
	[ħ]	Pharyngeal fricative
i	[ɨ]	Short, central, unrounded vowel
î	[i]	Close, front, unrounded vowel
j	[ʒ]	Post-alveolar voiced fricative
k	[k]	Velar voiceless unaspirated stop
	[kh]	Velar voiceless aspirated stop
l	[l]	Alveo-dental lateral
m	[m]	Bilabial nasal stop
n	[n]	Alveo-dental nasal stop
o	[o]	Close-mid, back, rounded vowel
p	[p]	Bilabial voiceless unaspirated stop
	[ph]	Bilabial voiceless aspirated stop
q	[q]	Uvular voiceless stop
r	[r]	Alveolar trill
	[ɾ]	Alveolar flap
s	[s]	Alveolar voiceless fricative
ş	[ʃ]	Post-alveolar voiceless fricative
t	[t]	Alveo-dental voiceless unaspirated stop
	[th]	Alveo-dental voiceless aspirated stop
u	[ʊ]	Near-close, slightly centralized rounded vowel
û	[u]	Close, back, rounded vowel
v	[v]	Labiodental voiced fricative
w	[w]	Bilabial approximant
x	[x]	Velar voiceless fricative
y	[j]	Palatal approximant
z	[z]	Alveolar voiced fricative

Table 2. Dataset from male speakers.

Male Speakers	Phonemes	Vowel Phonemes	Consonant Phonemes
Male 1	1332	583	749
Male 2	1096	469	627
Male3	1100	462	638
Male4	327	146	181
Total	3855	1660	2195

Table 3. Dataset from female speakers.

Female Speakers	Phonemes	Vowel Phonemes	Consonant Phonemes
Female 1	780	326	454
Female 2	1410	615	795
Female3	774	341	433
Total	2964	1282	1682

2.3. Proposed Model

In this study, the focus was placed on the recently suggested gated recurrent unit (GRU)-based model, the research on which had not been conducted for Kurdish speech before this study. The RNN was customized for speech segment detection based on the discrimination between consonants, vowels, and silence. Figure 1 shows the general steps of the proposed architecture for the Kurdish C/V/S speech segment detection.

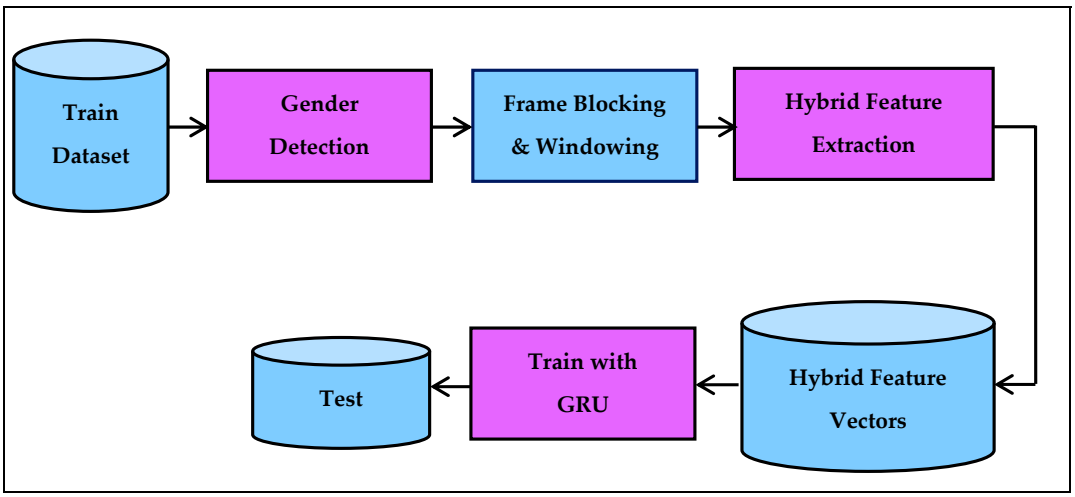


Figure 1. Flow chart of the proposed system. GRU: gated recurrent unit.

The training dataset contained C/V/S speech data that was tagged manually**.** The gender detection step identified the gender of a speaker from these tagged speech signals. In the frame blocking and windowing step, the tagged speech signals were segmented into small duration blocks of 20, 25, 30, and 35 ms known as frames. After that, each frame was multiplied with a Hamming, Hanning, or rectangular window function. In the hybrid feature extraction step, each windowed frame feature was obtained using "energy, ZCR, MFCC (EZMFCC)" or "energy, ZCR, D-MFCC (EZDMFCC)" or "energy, ZCR, DD-MFCC (EZDDMFCC)" techniques. In the hybrid feature vectors step, 15-, 28-, or 41-dimensional hybrid feature vectors were generated per frame. In the training with the GRU step, each different dimensional hybrid feature vectors were given to the GRU for learning a C/V/S pattern. The GRU computation was performed by the gated recurrent unit, which modulates information inside the unit without having a separate memory cell. It combines the "forget gate" and "input gate" into a single "update gate" and has an additional "reset gate."

A GRU network accepts an input sequence $x = (x_1; \ldots ; x_t)$. Each x_t represents a hybrid feature frame vector during the training process. The hybrid feature frame vector training set is introduced into the network as a GRU block series.

Each *j*.GRU block was computed in evolutionary terms based on the following equations in an iterative way from $t = 1$ to T.

At time *t*, the activation of the *j*.GRU output is the ${h_t}^j$.GRU unit.

The activation ${h_t}^j$ of the GRU at time *t* is a linear interpolation between the previous activation ${h_{t-1}}^j$ and the candidate activation $\tilde{h}_t^{\,j}$ are represented in Equations (1) and (2):

$$\tilde{h}_t{}^j = tanh\left(W \cdot \left[r_t{}^j * h_{t-1}{}^j, x_t\right]\right), \quad (1)$$

where $r_t{}^j$ refers to a set of reset gates, and $*$ refers to an element-wise multiplication.

$$h_t{}^j = (1 - z_t{}^j)h_{t-1}{}^j + z_t{}^j \tilde{h}_t{}^j. \quad (2)$$

The update gate $z_t{}^j$ decides how much of an update of its activation the unit performs and checks how much the past state must mean at the moment. There will be active reset gates r in units having short-term dependencies, while there are active update gates z in units having long-term dependencies. The update gates are given as Equation (3):

$$z_t{}^j = \sigma(W_Z x_t + U_Z h_{t-1)})^j, \quad (3)$$

In case of the off reset gate ($r_t{}^j = 0$), it ensures that the unit forgets the past. This is similar to allowing the unit to read the first symbol of an input sequence, where σ is the sigmoid function. W_z is the weight vector for the update gate, and W_r is the weight vector for the reset gate.

The reset gate is computed using Equation (4):

$$r_t{}^j = \sigma(W_r x_t + r h_{t-1})^j. \quad (4)$$

Hybrid feature frame vectors were utilized by GRU blocks in order to strongly match C/V/S patterns. Figure 2 describes the proposed GRU-trained model.

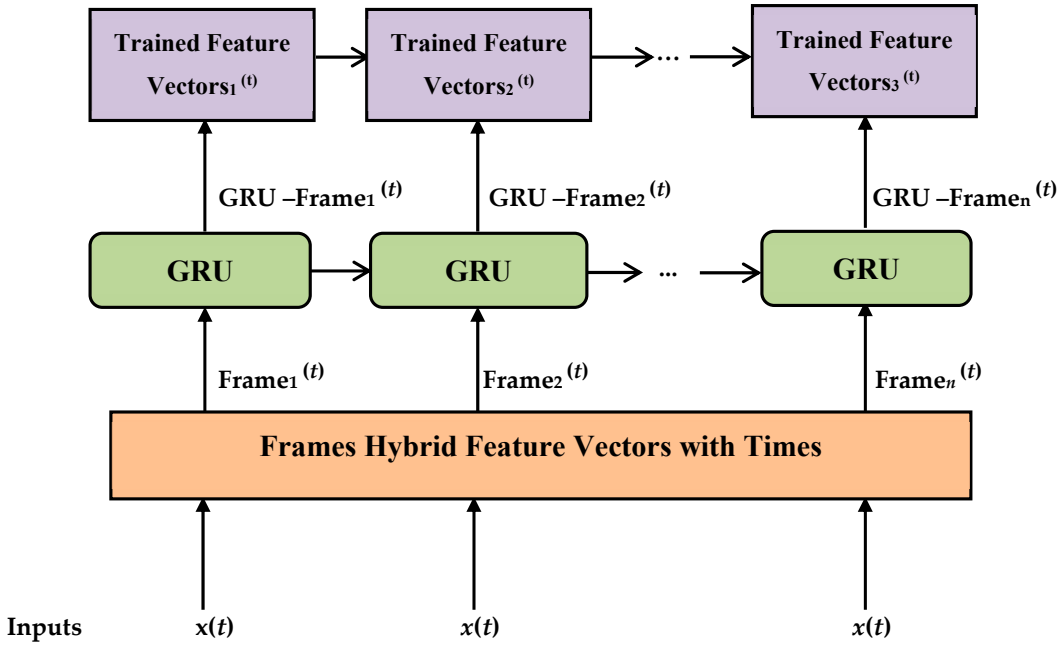


Figure 2. Proposed GRU-based framework.

The GRU-based trained feature vectors are used by CNN, MLP, naive Bayes, SVM, random forest, and k-NN classifiers in the testing step.

2.3.1. Created Tagged Speech

First, the voice data obtained in this study were subjected to a Wiener filter as a preprocessing step to eliminate noises from environmental factors and microphone noises [38]. The Wiener filter is a method performed in frequency space, where $y(n)$ is a noisy speech signal that is expressed using Equation (5):

$$y(n) = x(n) + v(n), \tag{5}$$

where $x(n)$ is the clean signal, $v(n)$ is the white Gaussian noise, and n is the discrete time variable.

The error signal ($e_x(n)$) between the clean speech sample at time n and its estimate can be defined using Equation (6):

$$e_x(n) \triangleq x_n - \mathbf{h}^T\mathbf{y}(n), \tag{6}$$

where the superscript ^{T} denotes the transpose of a vector or a matrix. $\mathbf{h}$ is a Finite Impulse Response (FIR) filter of length L and its transpose is given in Equation (7).

$$\mathbf{h}^T = \left[h_0, h_1, \ldots, h_{L-1,}\right] \tag{7}$$

$\mathbf{y}$ is a vector containing the most recent samples of the observation signal $y(n)$ and its transpose is given in Equation (8).

$$\mathbf{y}^T = [y(n), \ldots, y(n-1), y(n-L+1)] \tag{8}$$

The Wiener filter coefficients are obtained by minimizing the average squared error expressed by $e_x^2(n)$.

To create a dataset and assign an appropriate speech class (C/V/S) to each speech signal, every speech sentence should have an associated phoneme-level segmentation.

In this work, to create a reference dataset, manual segmentation methods were used. For this purpose, the "Pratt" package was used as a major tool. Every speech sentence was carefully examined and labeled manually using the selected tool. After labeling was completed, a tagged speech label file was created.

The manual segmentation method is considered to exhibit more accurate performance in comparison with automatic segmentation [39–41]. Therefore, manual segmentation was employed to create the unique labeled Kurdish dataset to be used in other studies in the future.

The phoneme segmentation is represented in Figure 3. In this figure, a sample annotated speech waveform and its spectrogram representing the Kurdish sentence "Bivir û hinçekên din" spoken by an adult female are shown.

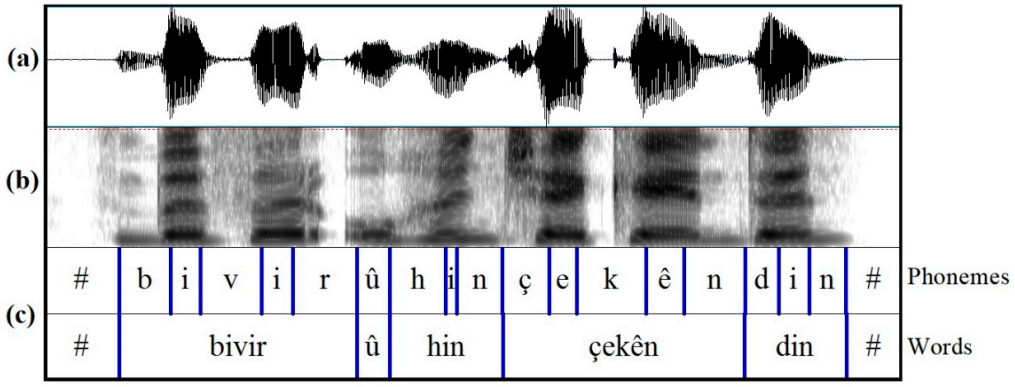


Figure 3. Speech waveform (**a**); spectrogram (**b**) and phonemes and words segmentation (**c**) of the Kurdish sentence.

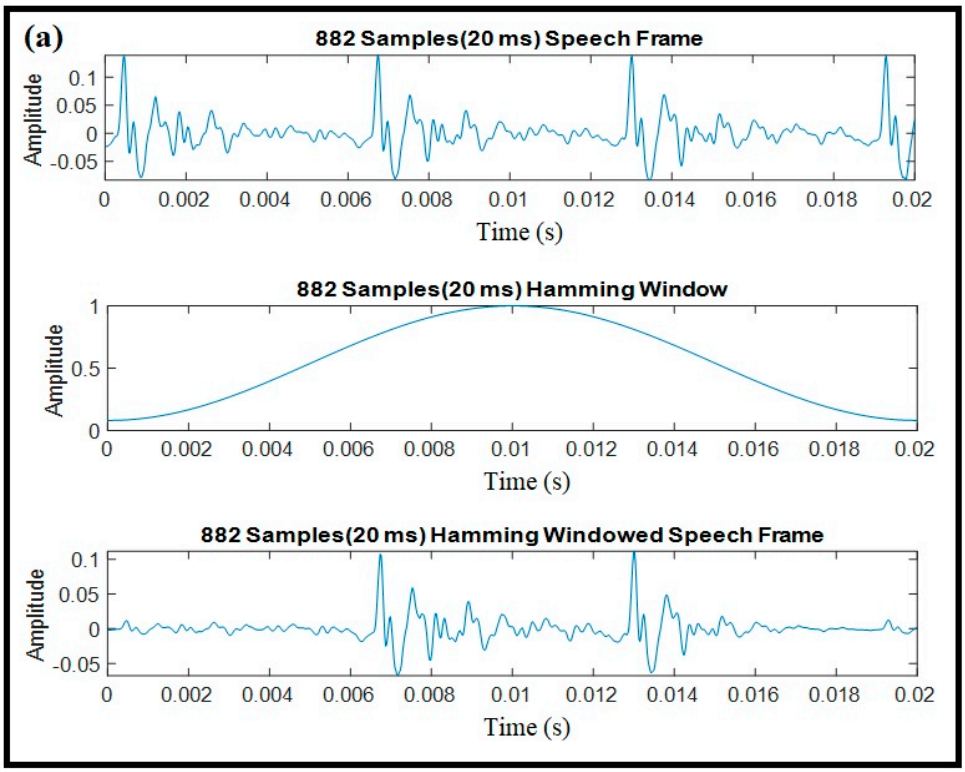
2.3.2. Gender Detection

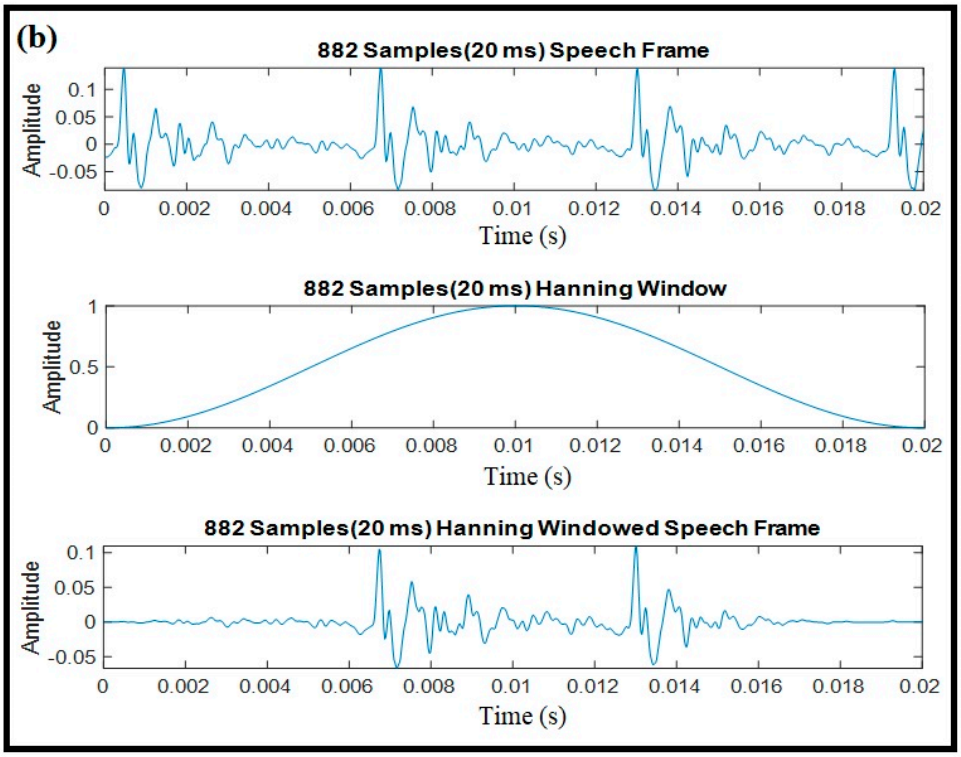
The short-time autocorrelation and average magnitude difference function (AMDF) was used collectively by assigning some weightage factor and setting a threshold based on the fact that males have a lower fundamental frequency of approximately 120 Hz compared to a female fundamental frequency of approximately 200 Hz [42].

2.3.3. Frame Blocking and Windowing

The speech signals were processed efficiently between short-time periods of 20 to 35 ms since distinctive features of sound signals were stable only within short periods of time. The speech data was divided into short periods of time intervals frame by frame.

Windowing was applied to framed speech signals. The purpose of this step was to eliminate discontinuities at the beginning and end of each frame [43]. This operation can be accomplished using a function that is called the window function. The most commonly used windowing functions are the Hamming, Hanning, and rectangular windowing techniques [44,45]. Windowing examples with these functions for 20 ms speech recorded from a female speaker are illustrated in Figure 4.





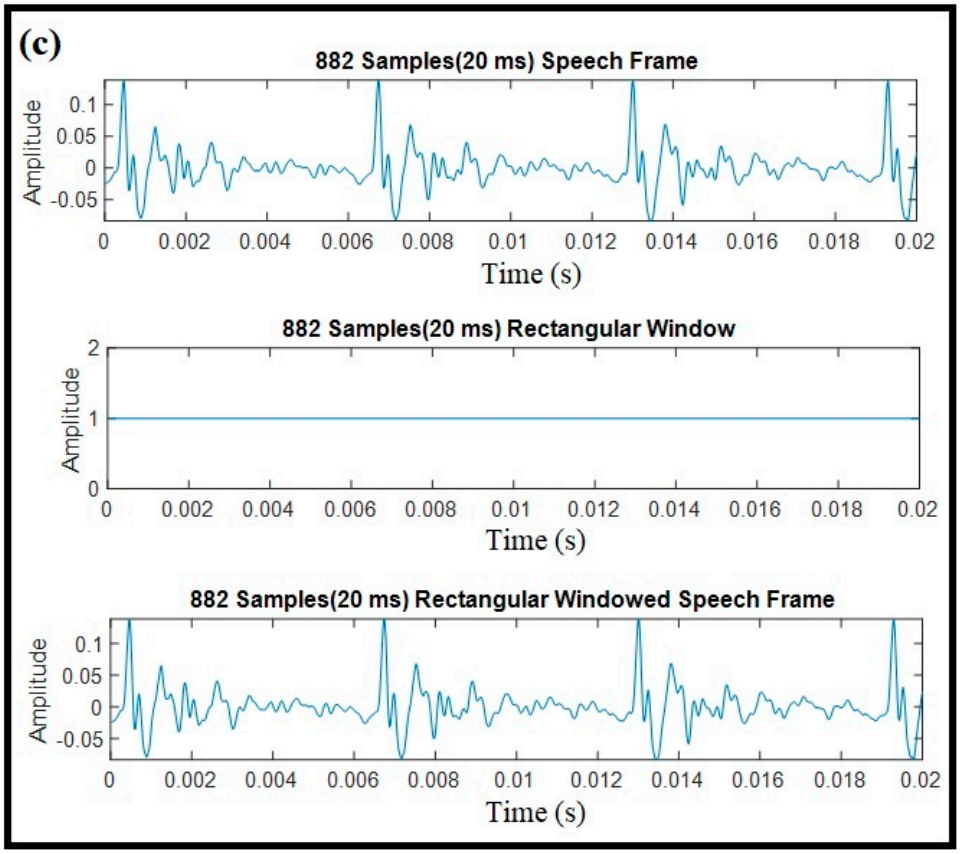


Figure 4. Examples of (**a**) Hamming, (**b**) Hanning, and (**c**) rectangular windows.

A windowing operation is simply the multiplication of the speech data and windowing function for each data frame. For the purpose of performance analysis, three window functions were implemented and the corresponding outputs are indicated.

2.3.4. Hybrid Feature Extraction

1. Short-Time Signal Energy

Short-time energy represents the dominant and most natural characteristic utilized. From a physical aspect, energy constitutes a measure of how much signal there is at any moment. The calculation of a signal's energy is generally performed on a short-term basis as a result of windowing the signal at a specific time, squaring samples, and obtaining the average [46–48]. The square root of the signal's energy is known as the root mean square (RMS) value. The short-time energy function (E_n) of a speech frame having length N is described in Equation (9):

$$E_n = \frac{1}{N}\sum_{m=-\infty}^{\infty}[x(m)w(n-m)]^2 . \quad (9)$$

The RMS energy ($E_{n(\mathrm{RMS})}$) of the frame in question can be presented as in Equation (10):

$$E_{n(RMS)} = \sqrt{\sum_{m=-\infty}^{\infty}[x(m)w(n-m)]^2}, \quad (10)$$

where $x(m)$ denotes the discrete-time audio signal, n is the time index of the short-time energy, and $w(m)$ denotes the window function.

2. Short-Time Zero Crossing Rate

The ZCR denotes the number of times speech samples alter the algebraic sign in a specific frame. The rate indicating the occurrence of zero crossings represents a simple measure of the signal's frequency content. It is also a measure of the number of times in a particular time interval/frame when the amplitude of speech signals passes through a zero value [46–48].

The short-time zero crossing (Z_n) of a speech frame and $sgn[x(n)]$ are described in Equations (11) and (12), respectively:

$$Z_n = \frac{1}{2}\sum_{m=-\infty}^{\infty}|sgn[x(n-m)] - sgn[x(n-m-1)]\,|w(m), \quad (11)$$

Where:

$$sgn[x(n)] = \begin{cases} 1, x(n) \geq 0 \\ -1, x(n) < 0' \end{cases} \quad (7)$$

and $w(m)$ refers to a window with length n, given in the equation.

3. Mel Frequency Cepstral Coefficient (MFCC)

MFCC is one of the most widely used and most effective feature extraction methods in most speech processing systems. One of its significant advantages is that the same words are not affected too much by changes that occur during vocalization.

MFCC imitates the perception of human ears and is calculated by using a fast Fourier transform (FFT), which is a fast algorithm used to implement the discrete Fourier transforms. The discrete Fourier transform (DFT) is expressed as in Equation (13) for an N-sample frame:

$$f(n) = \sum_{k=0}^{N-1} y_k\, e^{-\frac{2\pi jkn}{N}}, 0 \leq n \leq N-1, \quad (8)$$

where *f(n)* denotes the DFT, n is the sample index, and y refers to the signal windowing results.

Since the human hearing system perceives frequency values up to 1 kHz linearly and frequency values higher than 1 kHz logarithmically, there was a need for a unit that takes the hearing system as a model [49]. This unit is called the Mel frequency, where linear frequency values are converted into Mel frequency values using Equation (14), where "f" refers to the frequency in Hz, and "Mel (f)" refers to the Mel frequency:

$$\text{Mel}(f) = 2595 \times log\left(1 + \frac{f}{700}\right). \quad (9)$$

Finally, the Mel spectrum, of which the logarithm has been taken, is converted back to the time domain.

Delta-MFCC (D-MFCC) and the delta-delta-MFCC (DD-MFCC) are the first-order and the second-order derivatives of the MFCC, respectively [50]. D-MFCC and DD-MFCC are also known under the name of differential and acceleration coefficients.

The D-MFCC coefficients are calculated using Equation (15):

$$d_t = \frac{\sum_{\theta=1}^{\Theta} \theta(c_{t+\theta} - c_{t-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2}, \quad (10)$$

where d_t refers to the delta coefficient at time t, calculated with regard to the corresponding static coefficients $c_{t-\theta}$ to $c_{t+\theta}$, and Θ refers to the size of the delta window.

The DD-MFCC coefficients are computed by taking derivative of Equation (15).

2.3.5. Gated Recurrent Unit Recurrent Neural Networks

The gated recurrent unit (GRU) represents a kind of recurrent neural network. The GRU offers comparable performance and is significantly faster to compute. The network learns how to use its gates to protect its memory such that it is able to make longer-term predictions [51].

Figure 5 describes the architecture of a recurrent neural network, where x_t and S_t denote an input and hidden units for a given time step, respectively, at time t. O_t is the output at step t. U, W, and V are weight vectors for inputs, hidden layers, and outputs, respectively. The GRU recurrent neural network is illustrated in Figure 6, where z and r represent the update and reset gate vectors, respectively. h and $\tilde{h}$ are the activation and the candidate activation. The update and reset gates ensure that the cell memory is not taken over by tracking short-term dependencies. This means that we have dedicated mechanisms for when the hidden state should be updated and when it should be reset. The decision on what information is thrown away or kept in the cell is made by the sigmoid layer.

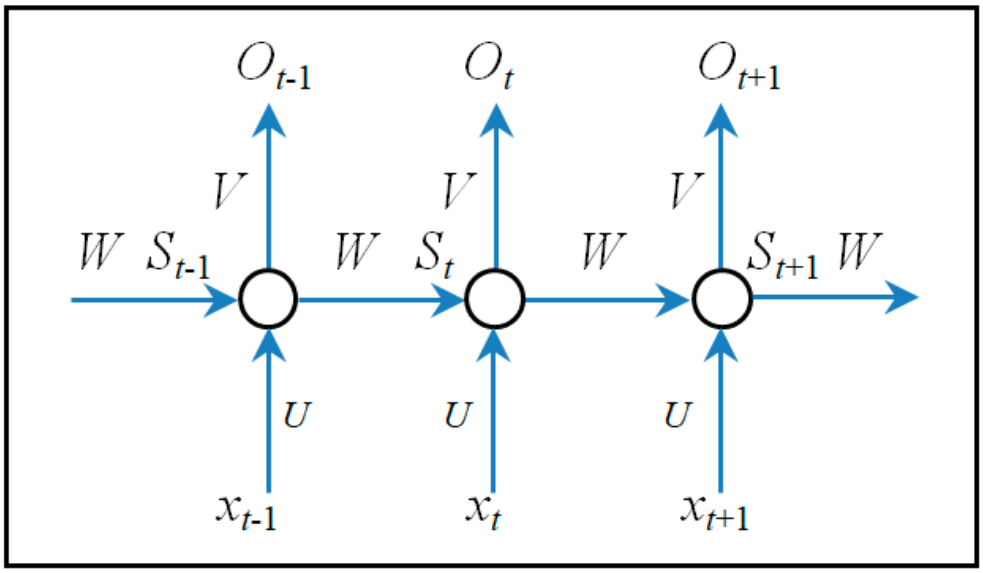


Figure 5. The architecture of recurrent neural networks [51].

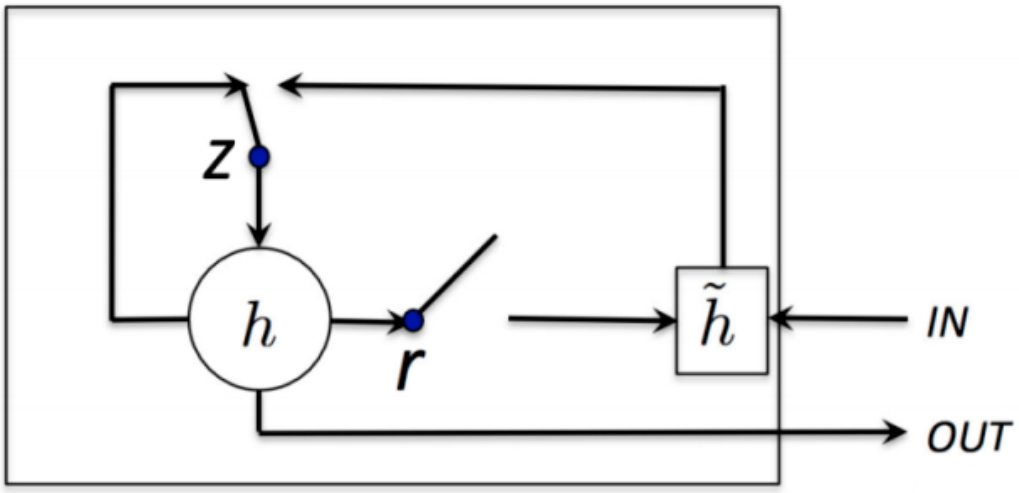


Figure 6. The architecture of a GRU recurrent neural network [51].

2.3.6. Artificial Neural Network (ANN)

1. Multilayer Perceptron (MLP)

A multilayer perceptron is a thinking structure that is formed by connecting neurons to each other with synaptic connections, which is inspired by the human brain and has a learning algorithm, similar to neural networks in biological systems. An MLP represents a feed-forward artificial neural network (ANN) model, which maps sets of input data onto a set of suitable outputs. Figure 7 presents the architecture of MLP. An MLP contains multiple layers of nodes in a directed graph, with every layer completely connected to the following one. Apart from the input nodes, every node constitutes a neuron, or a processing element, with a nonlinear activation function. The MLP uses a supervised learning technique that is called back-propagation in order to train the network [52].

2. Convolutional Neural Network (CNN).

Deep learning is generally applied by utilizing the neural network architecture. With deep learning, the model learns and abstracts the relevant information automatically as the data passes through the network. The term "deep" denotes the number of layers in the network, i.e., the higher the number of layers is, the deeper the network is. Layers are interconnected through nodes, or neurons, with every hidden layer by utilizing the previous layer's output as its input.

The most popular deep learning network is a convolutional neural network (CNN, or ConvNet). The CNN includes an input layer, an output layer, and several hidden layers. Convolution, activation or using a rectified linear unit (ReLU), and pooling represent the three most frequent layers. The above-mentioned operations are performed repeatedly over tens or hundreds of layers, with every layer learning for the purpose of detecting various features in the input data [53]. Figure 8 shows the general framework of convolution neural networks (CNNs).

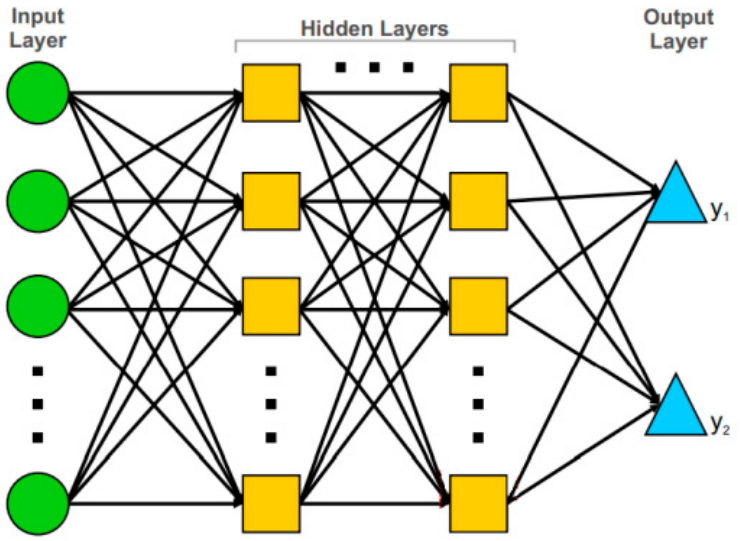


Figure 7. The architecture of a multilayer perceptron (MLP) [53].

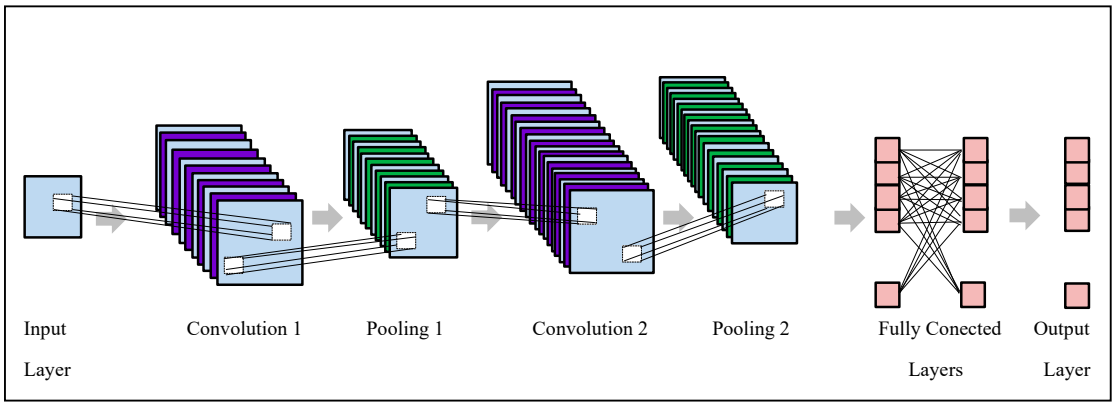


Figure 8. The architecture framework of a convolutional neural network [53].

3. Results and Discussion

In this study, we used three types of hybrid feature vectors. The first type of hybrid feature vector comprises EZMFCC parameters. The second type of hybrid feature vector comprises EZDMFCC parameters. The third type of hybrid feature vector comprises EZDDMFCC parameters. The above-mentioned types of hybrid features were extracted for each 20, 25, 30, and 35 ms window duration using the Hamming, Hanning, and rectangular windowing functions. All these parameters

were used at the feature extraction stage in the speech segment detection. The hybrid feature extraction process produced hybrid feature vectors, which were introduced as an input into the GRU system for learning a pattern and finding an optimal solution for each speech category (C/V/S). After the GRU was trained with the training set, an optimal hybrid feature parameter was found using the EZDDMFCC. Different experiments have been conducted for the C/V/S segment detection with different classifier models. The most common classifier models are CNN, MLP, naive Bayes, SVM, random forest, and k-NN. For the k-NN, the parameter "*k*" was set to 3. For the MLP, the number of hidden neurons suggested by the empirical selection was 10, 20, 30, 40, and 50 hidden neurons, which were trained separately, and the performance of every neuron was assessed. For the CNN, two convolutional layers with 20 and 100 feature maps with 5 × 5 patch sizes and 2 × 2 max pooling, respectively, were used.

Weka was utilized for obtaining classification accuracies. Accuracy was taken as the performance measure when counting the number of correct C/V/S speech segment detections. The accuracy function was calculated using Equation 16:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}, \tag{11}$$

where *TP*, *TN*, *FP*, and *FN* are true positive, true negative, false positive, and false negative, respectively.

Table 4–6 present the accuracy results obtained with the proposed GRU-based training method for male speakers, while Table 7–9 present the accuracy results obtained with the proposed GRU-based training method for female speakers.

3.1. Results of the Analysis of the Hybrid Features with GRU-Based Training

According to the results presented in the tables, the best accuracy was obtained using EZDDMFCC for all window types, window sizes, classification methods, and speaker gender. This might be due to the number of novel features existing in EZDDMFCC as a result of double derivation of the MFCC. The feature dimension of EZDDMFCC was higher than the others. However, it took longer time to compute, as seen in Table 10.

3.2. Results of the Analysis of the Window Size with GRU-Based Training

The experiment conducted on 20 ms, 25 ms, 30 ms, and 35 ms of speech segments gave the maximum performance accuracies of 97.60%, 96.38%, 96.60%, and 97.12%, respectively, for male speakers, and the maximum performance accuracies of 95.30%, 95.09%, 95.03%, and 95.63%, respectively, for female speakers. According to the results, the effect of window size on the accuracy was insignificant. However, a window size of 20 ms performed slightly better than the others. Actually, performance of the window size depended on the nature of the speech signal.

3.3. Results of the Analysis of the Window Type with GRU-Based Training

The results based on three different window types gave the highest accuracy values of 97.25% for the Hamming windows, 97.04% for the Hanning windows, and 97.60% for the rectangular windows for male speakers, and 95.10% for the Hamming windows, 95.63% for the Hanning windows, and 95.30% for the rectangular windows for female speakers. According to the results, the effect of the window type on the accuracy was insignificant.

3.4. Results of the Analysis of the Classification Model with GRU-Based Training

According to the results, the CNN showed a relatively high classification performance (97.60% for male and 95.63% for female speakers) compared to others.

Table 4. Performance accuracy for male speakers based on GRU+EZMFCC.

Windows		Classification Type and Performance (%)					
Type	**Size (ms)**	**CNN**	**MLP**	**Naive Bayes**	**SVM**	**Random Forest**	**k-NN**
Hamming	20	92.60	84.57	74.37	83.14	85.56	84.94
	25	92.11	84.19	74.63	82.91	86.36	85.42
	30	92.04	84.54	73.80	82.93	86.41	84.08
	35	92.59	84.12	74.66	83.36	86.37	84.43
Hanning	20	92.59	84.40	74.51	83.16	85.41	84.50
	25	92.40	84.38	74.61	83.14	86.10	84.89
	30	92.18	85.11	74.48	83.20	86.36	84.72
	35	92.80	85.38	74.51	83.38	87.10	83.42
Rectangular	20	92.05	85.24	73.80	83.30	85.63	84.49
	25	92.24	85.26	74.70	83.38	86.25	83.70
	30	92.44	85.41	74.01	83.66	86.43	83.64
	35	92.85	84.32	74.12	83.33	86.56	83.42

Table 5. Performance accuracy for male speakers based on GRU+EZDMFCC.

Windows		Classification Type and Performance (%)					
Type	**Size (ms)**	**CNN**	**MLP**	**Naive Bayes**	**SVM**	**Random Forest**	**k-NN**
Hamming	20	93.98	88.52	77.76	85.20	89.23	86.71
	25	94.71	88.37	78.49	85.44	88.28	85.01
	30	94.44	88.51	77.24	86.15	88.38	84.58
	35	94.00	87.65	77.71	86.03	89.14	84.17
Hanning	20	93.33	88.08	77.75	86.39	88.35	86.19
	25	94.00	87.72	78.22	86.30	88.89	87.03
	30	94.55	88.63	77.29	85.01	88.09	86.79
	35	94.21	87.70	77.01	85.47	88.23	86.29
Rectangular	20	94.61	86.97	77.35	85.55	88.82	85.18
	25	93.74	88.05	77.73	86.19	88.43	84.54
	30	93.79	88.13	77.08	84.62	87.62	84.61
	35	94.18	87.23	77.55	84.03	88.12	85.31

Table 6. Performance accuracy for male speakers based on GRU+EZDDMFCC.

Windows		Classification Type and Performance (%)					
Type	**Size (ms)**	**CNN**	**MLP**	**Naive Bayes**	**SVM**	**Random Forest**	**k-NN**
Hamming	20	97.25	91.14	82.20	89.06	93.53	90.38
	25	95.77	92.05	82.28	90.32	92.14	90.44
	30	96.53	92.36	81.30	89.80	93.19	90.87
	35	97.12	91.28	82.45	89.29	92.70	90.10
Hanning	20	97.04	92.65	81.80	89.81	93.74	90.49
	25	96.18	92.76	81.65	89.63	93.55	91.52
	30	96.60	92.13	80.28	89.04	92.00	90.24
	35	96.69	91.05	80.30	90.13	91.87	89.64
Rectangular	20	97.60	91.94	82.71	88.88	92.32	91.12
	25	96.38	92.42	81.73	89.37	93.15	90.23
	30	96.22	92.08	81.45	88.17	92.03	90.61
	35	95.98	92.32	81.38	88.85	92.32	90.33

Table 7. Performance accuracy for female speakers based on GRU+EZMFCC.

Windows		Classification Type and Performance (%)					
Type	**Size (ms)**	**CNN**	**MLP**	**Naive Bayes**	**SVM**	**Random Forest**	**k-NN**
Hamming	20	91.61	83.86	71.01	81.57	85.08	84.35
	25	91.11	82.34	72.11	81.43	84.97	83.22
	30	91.20	83.18	72.76	80.91	84.13	84.43
	35	91.14	82.12	72.93	80.79	84.11	83.07
Hanning	20	91.00	82.10	71.58	80.1306	84.76	84.32
	25	91.34	82.70	71.19	80.7541	86.30	85.14
	30	91.66	82.03	72.77	81.2109	85.14	84.22
	35	91.31	82.10	71.91	80.9700	86.02	83.36
Rectangular	20	91.22	82.16	71.02	80.8215	86.12	85.54
	25	91.01	82.30	71.46	80.6337	85.00	85.47
	30	91.41	82.57	72.18	81.0198	84.22	84.72
	35	91.60	83.70	72.29	80.9412	84.16	83.71

Table 8. Performance accuracy for female speakers based on GRU+EZDMFCC.

Windows		Classification Type and Performance (%)					
Type	**Size (ms)**	**CNN**	**MLP**	**Naive Bayes**	**SVM**	**Random Forest**	**k-NN**
Hamming	20	93.14	85.79	73.28	84.54	88.29	87.40
	25	93.37	87.31	73.44	85.63	88.41	85.64
	30	92.15	86.76	74.12	84.52	88.70	86.81
	35	93.50	87.61	74.59	84.25	89.57	86.26
Hanning	20	93.01	85.28	74.25	84.31	90.45	87.52
	25	93.30	86.39	74.02	84.29	88.07	86.89
	30	93.08	87.20	73.20	84.51	88.42	86.18
	35	92.51	87.28	73.57	84.44	89.90	86.87
Rectangular	20	92.26	87.11	74.18	84.25	88.57	87.83
	25	93.18	86.04	74.15	84.34	90.20	87.63
	30	93.88	86.57	74.82	84.56	89.27	87.60
	35	93.52	86.13	74.64	85.32	88.82	86.47

Table 9. Performance accuracy for female speakers based on GRU+EZDDMFCC.

Windows		Classification Type and Performance (%)					
Type	**Size (ms)**	**CNN**	**MLP**	**Naive Bayes**	**SVM**	**Random Forest**	**k-NN**
Hamming	20	95.10	91.30	75.51	86.29	92.20	91.16
	25	95.09	90.61	78.84	86.17	92.36	91.27
	30	95.03	90.87	77.21	86.94	91.44	91.35
	35	94.67	90.12	77.54	86.75	91.78	91.41
Hanning	20	94.82	89.48	78.31	86.01	92.12	91.64
	25	94.31	91.85	76.33	87.20	91.05	92.30
	30	94.93	91.70	77.82	86.76	92.80	91.54
	35	95.63	90.62	77.77	86.10	92.44	91.30
Rectangular	20	95.30	91.47	78.60	87.27	92.30	91.30
	25	94.16	90.56	79.80	87.05	92.49	92.20
	30	94.14	91.57	78.01	87.37	91.33	91.73
	35	95.05	91.13	77.12	87.53	92.70	92.17

Figures 9–11 show the accuracy results obtained with and without the proposed GRU-based training method for different hybrid features, window sizes, and window types. Only the CNN was employed as a classifier since it outperformed the others. The results demonstrate that the best performance was achieved with EZDDMFCC (Figure 10). According to the accuracy values presented in Figure 10, the best performances for male and female speakers were obtained with the GRU rectangular window with a size of 20 ms and Hanning window with a size of 35 ms, respectively.

Algorithms were implemented on a PC with an Intel Core i3-2367M processor, a 4 GB memory size, and a 1.40 GHz clock speed. Table 10 summarizes the computational times of the implemented algorithms.

In this respect, it is possible to state that: computational complexity/accuracy of the EZMFCC < computational complexity/accuracy of the EZDMFCC < computational complexity/accuracy of the EZDDMFCC.

Table 10. Computational times (in seconds) for male speakers based on a 20-ms Hamming window with GRU. CNN: Convolutional neural network; MLP: Multilayer perceptron; SVM: Support vector machine; k-NN: k nearest neighbor; EZMFCC: energy, zero crossing rate (ZCR), and MFCC; EZDMFCC: energy, ZCR, and delta-MFCC; EZDDMFCC: energy, ZCR, and delta-delta-MFCC

Classification Types	EZMFCC	EZDMFCC	EZDDMFCC
CNN	25.50	43.73	67.66
MLP	25.25	43.26	65.82
Naive Bayes	25.20	43.21	64,81
SVM	25.21	43.22	64.83
Random Forest	25.25	43.28	64.89
k-NN	25.20	43.20	64.80

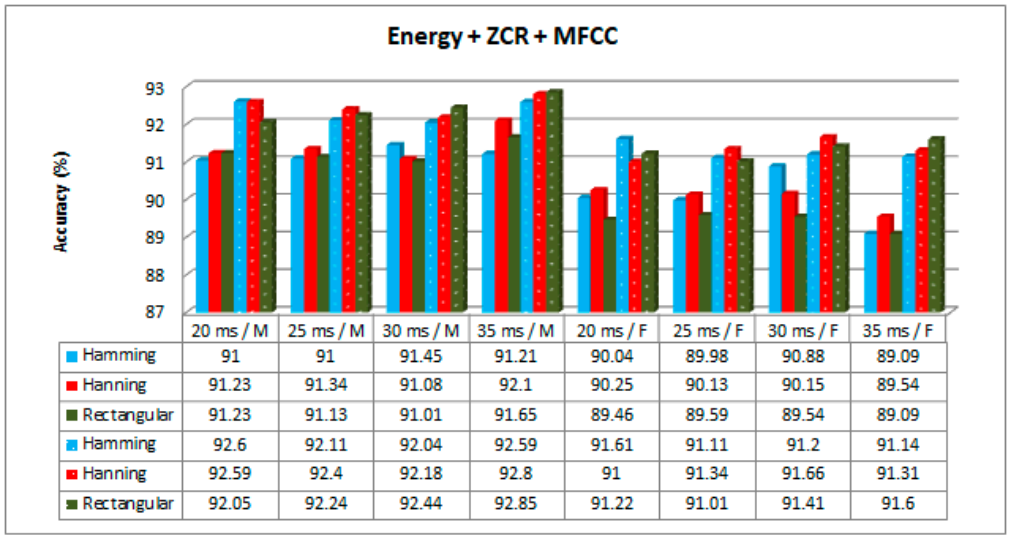
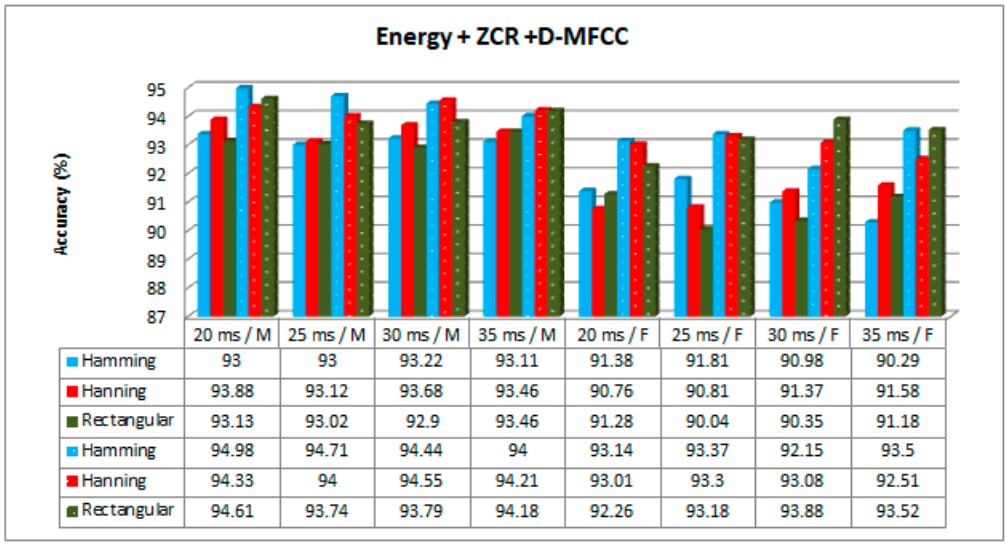
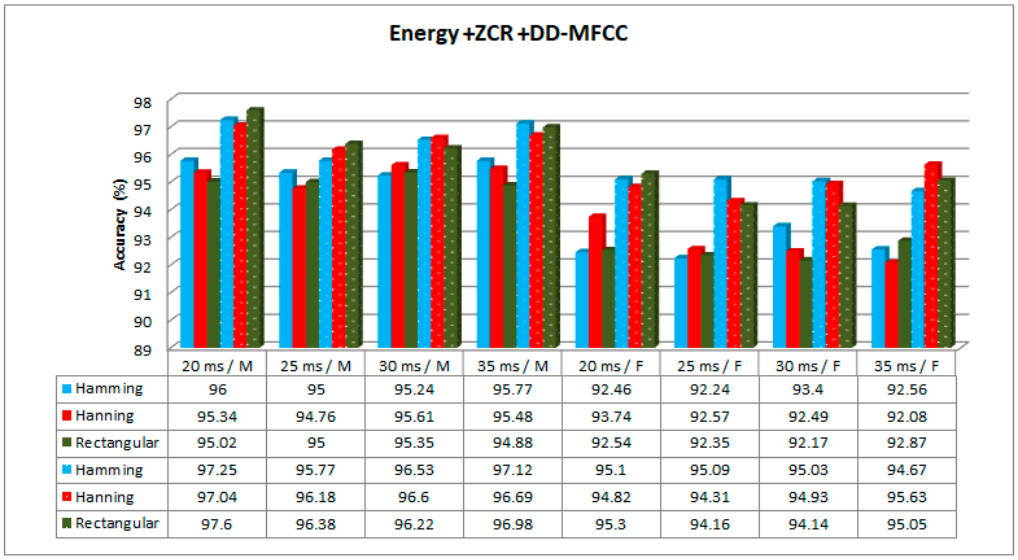


Figure 9. EZMFCC accuracy results without GRU-based training (no dotted bars) and with GRU-based training (dotted bars) using a CNN classifier for male (M) and female (F) speakers.



	20 ms / M	25 ms / M	30 ms / M	35 ms / M	20 ms / F	25 ms / F	30 ms / F	35 ms / F
Hamming	93	93	93.22	93.11	91.38	91.81	90.98	90.29
Hanning	93.88	93.12	93.68	93.46	90.76	90.81	91.37	91.58
Rectangular	93.13	93.02	92.9	93.46	91.28	90.04	90.35	91.18
Hamming	94.98	94.71	94.44	94	93.14	93.37	92.15	93.5
Hanning	94.33	94	94.55	94.21	93.01	93.3	93.08	92.51
Rectangular	94.61	93.74	93.79	94.18	92.26	93.18	93.88	93.52

Figure 10. EZDMFCC accuracy results without GRU-based training (no dotted bars) and with GRU-based training (dotted bars) using a CNN classifier for male (M) and female (F) speakers.



	20 ms / M	25 ms / M	30 ms / M	35 ms / M	20 ms / F	25 ms / F	30 ms / F	35 ms / F
Hamming	96	95	95.24	95.77	92.46	92.24	93.4	92.56
Hanning	95.34	94.76	95.61	95.48	93.74	92.57	92.49	92.08
Rectangular	95.02	95	95.35	94.88	92.54	92.35	92.17	92.87
Hamming	97.25	95.77	96.53	97.12	95.1	95.09	95.03	94.67
Hanning	97.04	96.18	96.6	96.69	94.82	94.31	94.93	95.63
Rectangular	97.6	96.38	96.22	96.98	95.3	94.16	94.14	95.05

Figure 11. EZDDMFCC accuracy results without GRU-based training (no dotted bars) and with GRU-based training (dotted bars) using a CNN classifier for male (M) and female (F) speakers.

In this study we used various combination parameters, and the models are summarized in Table 11.

Table 11. Feature parameter sets and models.

GRU Based Training Model													
Male and Female													
Hybrid Features	Classification Types	Hamming				Hanning				Rectangular			
		20	25	30	35	20	25	30	35	20	25	30	35
EZMFCC	CNN	√	√	√	√	√	√	√	√	√	√	√	√
	MLP	√	√	√	√	√	√	√	√	√	√	√	√
	Naïve Bayes	√	√	√	√	√	√	√	√	√	√	√	√
	SVM	√	√	√	√	√	√	√	√	√	√	√	√

	Random Forest	√	√	√	√	√	√	√	√	√	√	√	√
	k-NN	√	√	√	√	√	√	√	√	√	√	√	√
EZDMFCC	CNN	√	√	√	√	√	√	√	√	√	√	√	√
	MLP	√	√	√	√	√	√	√	√	√	√	√	√
	Naïve Bayes	√	√	√	√	√	√	√	√	√	√	√	√
	SVM	√	√	√	√	√	√	√	√	√	√	√	√
	Random Forest	√	√	√	√	√	√	√	√	√	√	√	√
	k-NN	√	√	√	√	√	√	√	√	√	√	√	√
EZDDMFCC	CNN	√	√	√	√	√	√	√	√	√	√	√	√
	MLP	√	√	√	√	√	√	√	√	√	√	√	√
	Naïve Bayes	√	√	√	√	√	√	√	√	√	√	√	√
	SVM	√	√	√	√	√	√	√	√	√	√	√	√
	Random Forest	√	√	√	√	√	√	√	√	√	√	√	√
	k-NN	√	√	√	√	√	√	√	√	√	√	√	√

No GRU Based Training Model

Male and Female

Hybrid Features	Classification Types	Hamming				Hanning				Rectangular			
		20	25	30	35	20	25	30	35	20	25	30	35
EZMFCC	CNN	√	√	√	√	√	√	√	√	√	√	√	√
EZDMFCC		√	√	√	√	√	√	√	√	√	√	√	√
EZDDMFCC		√	√	√	√	√	√	√	√	√	√	√	√

Various studies have been carried out to investigate speech segmentation. However, although there are studies conducted with standard datasets created by various universities and institutions in different languages, no comprehensive study to determine the optimum feature parameters based on the GRU for Kurdish speech segment detection has been encountered. The Kurdish vocabulary dataset, which was derived from TRT Kurdî Nûçe news speech signals, was used to train the classifier with GRU recurrent neural networks.

In this study the automatic detection of the C/V/S phoneme segmentation method used for the continuous Kurdish speech segmentation was discussed. The results demonstrate that the suggested method yields promising results for identifying the optimal set of feature parameters for the Kurdish language classifier. According to the obtained analysis results, it was observed that the selection of feature extraction methods and the selection of classifier models in male and female speakers were the two main factors affecting the performance of the Kurdish C/V/S speech detection system. For male voices, the optimal choice was the use of the rectangular window with a window size of 20 ms with EZDDMFCC feature parameters and the CNN classifier. For female voices, the optimal choice was the use of the Hanning window with a window size of 35 ms with EZDDMFCC feature parameters and the CNN classifier. We saw better performances for the male class with respect to female. We do not think that the difference in the performances was caused by the imbalance of the two classes. It may be attributed to the difference in the speaking style of female speakers.

As a result, it was observed that as the number of novel features increased, the success rate increased for female and male speakers. The windowing time and windowing techniques had an insignificant and similar contribution. The results demonstrate that CNN deep learning classifiers achieved higher classification performance compared to MLP and standard classifiers.

4. Conclusions

In this study, speech segment detection for the Kurdish language and a unique related dataset were developed. Performances of Kurdish speech segment detection with and without GRU were evaluated for different parameters, including window type, window size, and hybrid feature vectors. Extensive research to identify the optimal set of feature parameters and the effect of the analysis based on GRU recurrent neural networks was undertaken. It was shown that the Kurdish speech segment detection with GRU recurrent neural networks achieved the best performance. In

this study, we achieved promising classification accuracy for Kurdish speech segment detection by optimizing processing parameters. The current study may be extended for the purpose of building a speech recognition system based on the individual consonant/vowel unit. In the future, we also aim to further enhance the mentioned results by using different deep learning algorithms, such as LSTM and their bidirectional variants.

Author Contributions: Conceptualization, Ö.B.D. and N.A; methodology, Ö.B.D. and N.A; software, Ö.B.D.; validation, Ö.B.D.; formal analysis, Ö.B.D.; investigation, Ö.B.D. and N.A.; resources, Ö.BD.; data curation, Ö.B.D.; writing—original draft preparation, Ö.B.D. and N.A.; writing—review and editing, N.A.; visualization, Ö.B.D..; supervision, N.A.; project administration, N.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: Gratitude is expressed to the TRT Kurdî Nûçe site.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sakran, A.E.; Abdou, S.M.; Hamid, S.E.; Rashwan, M.A. A review: Automatic Speech Segmentation. *IJCSMC* **2017**, *6*, 308–315.
2. Artuner, H. The Design and Implementation of a Turkish Speech Phoneme Clustering System. Ph.D. Thesis, Ankara University, Ankara, Turkey, 1994.
3. Sharma, U. Measurement of formant frequency for constant-vowel type Bodo words for acoustic analysis. In Proceedings of the 2014 International Conference on Data Mining and Intelligent Computing (ICDMIC), New Delhi, India, 5–6 September 2014; pp. 1–4.
4. Nazmy, T.M.; Gadallah, M.E.; Abdelhamid, A.A. A novel method for Arabic consonant/vowel segmentation using wavelet transform. *IJICIS* **2005**, *5*, 353–364.
5. Ravanelli, M.; Brakel, P.; Omologo, M.; Bengio, Y. Light Gated Recurrent Units for Speech Recognition. *IEEE Trans. Emerg. Top. Comput. Intell.* **2018**, *2*, 92–102.
6. Shewalkar, A.; Nyavanandi, D.; Ludwig, S.A. Performance Evaluation of Deep Neural Networks Applied to Speech Recognition: RNN, LSTM and GRU. *JAISCR* **2019**, *9*, 235–245.
7. Cernak, M.; Tong, S. Nasal Speech Sounds Detection Using Connectionist Temporal Classification. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5574–5578.
8. Zheng, C.; Wang, C.; Jia, N. An Ensemble Model for Multi-Level Speech Emotion Recognition. *Appl. Sci.* **2019**, *10*, 1–20.
9. Chen, Z.; Zhang, X.; Deng, J.; Li, J.; Jiang, Y.; Li, W.A. Practical Singing Voice Detection System Based on GRU-RNN. *CSMT* **2019**, *568*, 15–25.
10. Zyl van Vuuren, V.; ten Bosch, L.; Niesler, T. Unconstrained speech segmentation using deep neural networks. In Proceedings of the ICPRAM, Lisbon, Portugal, 10–12 January 2015; pp. 248–254.
11. Franke, J.; Mueller, M.; Hamlaoui, F.; Stueker, S.; Waibel, A. Phoneme boundary detection using deep bidirectional LSTMs. In Proceedings of the Speech Communication, 12. ITG Symposium, Paderborn, Germany, 5–7 October 2016; pp. 77–381.
12. Wang, Y.-H.; Chung, G.-T.; Lee, H.-Y. Gate activation signal analysis for gated recurrent neural networks and its correlation with phoneme boundaries. *arXiv* **2017**, arXiv:1703.07588v2.
13. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555v1.
14. Hochreiter, S.; Schmidhuber, J. Long short term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
15. Lee, Y.H.; Yang, J.Y.; Cho, C.; Jung, H. Phoneme segmentation using deep learning for speech synthesis. In Proceedings of the 1329 RACS, Honolulu, HI, USA, 9–12 October 2018; pp. 59–61.
16. Graves, A.; Schmidhuber, J. Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures. *Neural Netw.* **2005**, *18*, 602–610.
17. Weinstein, C.; McCandless, S.S.; Mondshein, L.; Zue, V. A system for acoustic-phonetic analysis of continuous speech. *IEEE Trans. Acoust. Speech Signal Process.* **1975**, *23*, 54–67.

18. Leung, H.C.; Glass, J.R.; Phillips, M.S.; Zue, V.W. Phonetic classification using multi-layer perceptrons. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Albuquerque, NM, USA, 3–6 April 1990; pp. 525–528.
19. Ali, A.M.A.; Van der Spiegel, J.; Mueller, P. Acoustic-phonetic features for the automatic classification of stop consonants. *IEEE Trans. Speech Audio Process.* **2001**, *9*, 833–841.
20. Natarajan, V.A.; Jothilakshmi, S. Segmentation of continuous speech into consonant and vowel units using formant frequencies. *Int. J. Comput. Appl.* **2012**, *56*, 24–27.
21. Ades, A.E. Theoretical notes vowels, consonants, speech and nonspeech. *Psychol. Rev.* **1977**, *84*, 524–530.
22. Ooyen, B.V.; Cutler, A.; Norris, D. Detection times for vowels versus consonants. In Proceedings of the 2nd European Conference on Speech Communication and Technology (EUROSPEECH), Genoa, Italy, 24–26 September 1991; pp. 1451–1454.
23. Suh, Y.; Lee, Y. Phoneme segmentation of continuous speech using Multilayer Perceptron. In Proceedings of Fourth International Conference on Spoken Language Processing, ICSLP, Philadelphia, PA, USA, 3–6 October 1996; pp. 1–4.
24. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, Massachusetts, London, England, 2016. Available online: http://www.deeplearningbook.org (accessed on 1 November 2019).
25. Ma, X.; Wu, Z.; Jia, J.; Xu, M.; Meng, H.; Cai, L. Study on Feature Subspace of Archetypal Emotions for Speech Emotion Recognition. *arXiv* **2016**, arXiv:1602.05875.
26. Li, C.; Ma, X.; Jiang, B.; Li, X.; Zhang, X.; Liu, X.; Cao, Y.; Kannan, A.; Zhu, Z. Deepspeaker: An end-to-end neural speaker embedding system. *arXiv* **2017**, arXiv:1705.02304.
27. Zhao, J.; Mao, X.; Chen, L. Speech emotion recognition using deep 1D & 2D CNN LSTM networks. *Biomed. Signal Process. Control* **2019**, *47*, 312–323.
28. Wang, D.; Wang, X.; LV, S. End-to-End Mandarin Speech Recognition Combining CNN and BLSTM. *Symmetry* **2019**, *11*, 1–19.
29. Keren, G.; Schuller, B. Convolutional RNN: An Enhanced Model for Extracting Features from Sequential Data. *arXiv* **2016**, arXiv:1602.05875.
30. Xu, H.; Zhang, X.; Jia, L. The extraction and simulation of mel frequency cepstrum speech parameters. In Proceedings of the International Conference on Systems and Informatics (ICSAI), Yantai, China, 19–20 May 2012; pp. 1765–1768.
31. Boersma, P. Praat, a system for doing phonetics by computer. Glot International 2001, 5 : 9 / 10, 341-345.
32. Charles, P.W.D. Project Title, GitHub repository, 2013.
Available online: https://github.com/charlespwd/project-title (accessed on 13 March 2018)
33. Frank, E.; Hall, M.A.; Witten, I. H. The Weka Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition, 2016.
34. Lang, S.; Bravo-Marquez, F.; Beckham, C.; Hall, M.; Frank, E. WekaDeeplearning4j: a Deep Learning Package for Weka based on DeepLearning4j, In Knowledge-Based Systems 2019, 178, 48-50. DOI: 10.1016/j.knosys.2019.04.013.
35. Thackston, W.M. *Kurmanji Kurdish—A Reference Grammar with Selected Readings*; Cambridge, Mass. : Harvard University 2006; pp. 1–90. Available online: http://bibpurl.oclc.org/web/36880 (accessed on 25 December 2019).
36. Khan, E.D.B.; Lescot, R. *Kürtçe Grameri*; Institut Kurde de Paris, Paris, France, 1990; pp. 1–13.
37. Gündoğdu, S. Remarks on vowels and consonants in Kurmanji. *J. Soc. Sci. Muş Alparslan* **2016**, *4*, 1–14.
38. Chen, J.; Benesty, J.; Huang, Y.; Doclo, S. New insights into the noise reduction wiener filter. *IEEE Trans. Audio Speech Lang. Process.* **2005**, *14*, 1218–1234.
39. Cosi, P.; Falavigna, D.; Omologo, M. A preliminary statistical evaluation of manual and automatic segmentation discrepancies. In Proceedings of the European Conference on Speech Communication and Technology (EUROSPEECH), Genova, Italy, 24–26 September 1991; pp. 693–696.
40. Cox, S.J.; Brady, R.; Jackson, P. Techniques for accurate automatic annotation of speech waveforms. In Proceedings of ICSLP, Sydney, Australia, 30 November–4 December 1998; pp. 1947–1950.
41. Ljolje, A.; Hirschberg, J.; Van Santen, J.P.H. Automatic Speech Segmentation for Concatenative Inventory Selection; Progress in Speech Synthesis; Springer: New York, NY, USA, 1997; pp. 305–311.
42. Jain, N.; Kaushik, D. Gender voice recognition through speech analysis with higher accuracy. In Proceedings of the 8th International Conference on Advance Computing and Communication Technology, At Panipat, Haryana, 15 November 2014; pp. 1–5.

43. Aydin, N.; Markus, H.S. Optimization of processing parameters for the analysis and detection of embolic signals. *Eur. J. Ultrasound* **2000**, *12*, 69–79.
44. Harris, F.J. On the use of windows for harmonic analysis with discrete Fourier transform. *Proc. IEEE* **1978**, *66*, 51–83.
45. Chithra, P.L.; Aparna, R. Performance analysis of windowing techniques in automatic speech signal segmentation. *Indian J. Sci. Technol.* **2015**, *8*, 1–29.
46. Zhang, T.; Kuo, C.C. Hierarchical classification of audio data for archiving and retrieving. In Proceedings of the ICASSP, Phoenix, AZ, USA, 15–19 March 1999; pp. 3001–3004.
47. Hemakumar, G.; Punitha, P. Automatic segmentation of Kannada speech signal into syllable and sub-words: Noised and noiseless signals. *Int. J. Sci. Eng. Res.* **2014**, *5*, 1707–1711.
48. Kalamani, M.; Valarmathy, S.; Anitha, S. Hybrid speech segmentation algorithm for continuous speech recognition. *Int. J. Appl. Inf. Commun. Eng.* **2015**, *1*, 39–46.
49. Sidiq, M.; Budi, W.T.A.; Sa'adah, S. Design and implementation of voice command using MFCC and HMMs method. In Proceedings of the ICoICT, Nusa Dua, Bali, 27–29 May 2015.
50. Hossan, M.A.; Memon, S.; Gregory, M.A. A novel approach for MFCC feature extraction. In Proceedings of the ICSPCS, Gold Coast, Australia, 13–15 December 2010.
51. Rana, R. Gated recurrent unit (GRU) for emotion classification from noisy speech. *arXiv* **2016**, arXiv:1612.07778v1.
52. Misra, P.; Giri, A. Review of System Identification Using Neural Network Techniques. *Int. J. Electr. Electron. Data Commun.* **2014**, *2*, 13–16.
53. Feltes, B.C.; Grisci, B.L.; Poloni, J.F.; Dorn, M. Perspectives and Applications of Machine Learning for Evolutionary Developmental Biology. *Mol. Omics* **2018**, *14*, 289–306.