mathematics

Article

Language Accent Detection with CNN Using Sparse Data from a Crowd-Sourced Speech Archive

Veranika Mikhailava [1,†], Mariia Lesnichaia [2,†], Natalia Bogach [2,*], Iurii Lezhenin [2,3], John Blake [1], and Evgeny Pyshkin [1,*]

[1] School of Computer Science and Engineering, The University of Aizu, Aizu-Wakamatsu 965-8580, Japan
[2] Institute of Computer Science and Technology, Peter the Great St. Petersburg Polytechnic University, 195251 St. Petersburg, Russia
[3] Speech Technology Center, 194044 St. Petersburg, Russia
[*] Correspondence: bogach@kspt.icc.spbstu.ru (N.B.); pyshe@u-aizu.ac.jp (E.P.)
[†] These authors contributed equally to this work.

Abstract: The problem of accent recognition has received a lot of attention with the development of Automatic Speech Recognition (ASR) systems. The crux of the problem is that conventional acoustic language models adapted to fit standard language corpora are unable to satisfy the recognition requirements for accented speech. In this research, we contribute to the accent recognition task for a group of up to nine European accents in English and try to provide some evidence in favor of specific hyperparameter choices for neural network models together with the search for the best input speech signal parameters to ameliorate the baseline accent recognition accuracy. Specifically, we used a CNN-based model trained on the audio features extracted from the Speech Accent Archive dataset, which is a crowd-sourced collection of accented speech recordings. We show that harnessing time–frequency and energy features (such as spectrogram, chromogram, spectral centroid, spectral rolloff, and fundamental frequency) to the Mel-frequency cepstral coefficients (MFCC) may increase the accuracy of the accent classification compared to the conventional feature sets of MFCC and/or raw spectrograms. Our experiments demonstrate that the most impact is brought about by amplitude mel-spectrograms on a linear scale fed into the model. Amplitude mel-spectrograms on a linear scale, which are the correlates of the audio signal energy, allow to produce state-of-the-art classification results and brings the recognition accuracy for English with Germanic, Romance and Slavic accents ranged from 0.964 to 0.987; thus, outperforming existing models of classifying accents which use the Speech Accent Archive. We also investigated how the speech rhythm affects the recognition accuracy. Based on our preliminary experiments, we used the audio recordings in their original form (i.e., with all the pauses preserved) for other accent classification experiments.

Keywords: NLP; automatic accent identification; convolutional neural networks (CNN); Mel-frequency cepstral coefficients (MFCC); amplitude mel-spectrogram; crowd-sourced data collection

MSC: 68T07; 68T10; 68T50

Citation: Mikhailava, V.; Leshnichaia, M.; Bogach, N.; Lezhenin, I.; Blake, J.; Pyshkin, E. Language Accent Detection with CNN Using Sparse Data from a Crowd-Sourced Speech Archive. *Mathematics* **2022**, *10*, 2913. https://doi.org/10.3390/math10162913

Academic Editor: Manuel Vilares-Ferro, Pavel Brazdil and Gaël Dias

Received: 31 May 2022
Accepted: 9 August 2022
Published: 13 August 2022

1. Introduction

Speech features associated with distinctive ways of pronunciation connected to the speaker's gender, age, family, social class, geographic location, and mother tongue are instantiated in the form of different language accents. Specifically, foreign accents can be considered as a compound effect of contact between two L1 and L2 phonological systems, where L1 is derived from the speaker's native language, while L2 refers to the second language [1]. As reported in [2], the accuracy of automatic speech recognition (ASR) word-processing software can be high for native-speaking users but drops significantly for L2 speakers with advanced level proficiency, but accented speech. Accent-aware modeling has been recently reported as an efficient approach to improve mispronunciation detection and diagnosis systems [3,4]. However, it is usually assumed that the information about the accent of an utterance is known in both the training and testing phase, though, in real life scenarios, the accent might be *a priori* unknown. Automatic foreign accent recognition (i.e., detection of the speaker's L1 based on L2 samples) can improve the robustness of ASR-based software and computer-assisted pronunciation training (CAPT) systems. Accent detection can contribute to overcoming the unwanted variability of speaker-independent speech recognition models [5–7]. Conventional acoustic language models adapted to fit the standard language corpus are unable to satisfy the recognition requirements for accented speech. Solving the problem of accented speech recognition by adding more pronunciation samples to the dataset used for training is inappropriate, since such an approach increases the processing time and creates additional noise that degrades performance [8].

It is known that speakers with heavy accents tend to make more errors in terms of standard L2 pronunciation. Experimental analysis has shown that this type of error makes up a significant percentage of the total number of speech errors in L2 pronunciation. In addition, speakers from regions with the same accent have been observed to have similar trends in mispronunciation [9]. Therefore, for systems that aim to provide feedback on L2 pronunciation, it makes sense to determine the speaker's speech accent in L2. The knowledge gained from accent classification can improve the overall performance of an ASR system and make it more reliable; since, in the case of preliminary accent identification, the speech recognizer can be further trained for a specific accent group [5,8,10].

Research shows accent detection can contribute to significant improvements in algorithms, models, and interfaces of other human-centric systems, including but not limited to:

- Analysis and modeling of speakers' variability in frame of speech recognition [9];
- Development of user interaction scenarios in video-games [11];
- Analysis of phonetic particularities and related personal behavior [12];
- Using accent-related information as components of biometric data [13];
- Mitigating accent influence in voice-control systems [14];
- Improving personalization of exercises and feedback in CAPT systems [2].

The remaining text is organized as follows. In Section 2, we describe the relevant research works contributing to the solution of the accent recognition problem, along with positioning our own work in the frame of the existing models. Section 3 introduces the methodology including CNN construction, accent detection, feature selection, data collection, model parameter classification, and the tools we used. In Section 4, the experimental results are presented across hyperparameter selection, regularization, and with respect to different sets of audio signal features used by the CNN classifier. Section 5 reports the evaluation approach using standard information retrieval metrics including accuracy, precision, recall, and F1. Section 6 discusses major experimental results and further possibilities for improvement, as well as potential areas of application. In the Conclusion, we summarize the major outcomes and findings of this work.

2. Scope of Research

In this part we delimit the accent groups we focus on, and analyze the existing models for accent classifiers and the input speech signal parameters regularly used in related works. We also extract the relevant research questions, such as inner model configuration, optimal feature set and rhythm impact. Table 1 lists the papers which are the closest to the scope of our study.

Table 1. Retrospective summary of related works.

Paper, Year	Feature Set	Model	Classes	Accents	Dataset
[15], 2022	Mel-spectrogram	CNN	5	5 Kashmiri accents	Custom
[16], 2021	SG	CNN (LeNet)	5	DU, FR, JA, NS, PO	IViE, Cambridge English Corpus
[12], 2021	SG	CNN	5	AR, FR, GE, IN, NS	SAA
[5], 2020	MFCC, SG, CG, SC, SR	CNN	5 3	AR, FR, NS, SP, ZH AR, NS, ZH	SAA
[17], 2020	MFCC	CNN with attention	2 4 9	IN, NS IN IN, NS	Custom
[18], 2020	MFCC	Logistic Regression	3	HA, IG, YO	Custom
[13], 2019	MFCC	LSTM, RF	4	NS, SP	Custom
[10], 2017	MFCC, LPCC	FFNN	6	GA, IN, IT, JA, KO, NS	Wildcat
[11], 2017	SG	CNN (AlexNet)	3	NS, SP	SAA
[19], 2017	MFCC	GMM	3	ML	Custom
[20], 2012	Mel-spectrogram statistics	FF-MLP	3	IN, MS, ZH	Custom
[8], 2005	2nd and 3rd formants	GMM	2	IN, NS	Custom (SAA subset)

As a sub-task of speech and language recognition, accent detection algorithms are built using the standard classification models and machine learning architectures including convolutional neural networks (CNN) [5,11,16,21], feedforward neural networks (FFNN) [10], hidden Markov model (HMM) [13], k-nearest neighbor (KNN) model [22], Gaussian mixture model (GMM) [23,24], long short-term memory (LSTM) and bidirectional LSTM (bLSTM) [25,26], random forest, and support vector machine (SVM) [13,22,24,27,28].

Accent classification accuracy is significantly affected by the input feature selection. As of 2021, the best experimental results have been achieved using mel-frequency cepstral coefficients (MFCC), along with other types of input features including spectrogram (SG), chromagram (CG), spectral centroid (SC), spectral rolloff (SR), and mel-weighted single filtered frequency (SFF) spectrogram [5,28].

Particularly, Singh, Pillay, and Jembere [5] managed to achieve a maximum accuracy of 53.92% while classifying five accents and 70.38% for three accents using mel-cepstral coefficients, extracted from three-word audio segments. The authors used the Speech Accent Archive dataset [29].

Based on the AlexNet architecture, Ensslin et al. [11] trained a tailored classification on three accents using 227 × 227 spectrogram images as input features applied to the same Speech Accent Archive dataset. The authors reported a CNN accuracy of 61% while recognizing among three English accents—namely, British, American and Spanish. In [17], Ahamad, Anand and Bhargava defined a list of requirements and collected a dataset conforming to these requirements to test a number of different classifiers including MLP, CNNs, and CNNs with an attention mechanism. Using MFCC as input features in CNNs with an attention mechanism showed the best accuracy with up to 100% for two classes, 99.0% for four classes, and 99.5% for nine classes.

The highest average recognition accuracy was achieved with the combination of MFCC and FFNN, thus, giving 91.43%, compared to 78.73% while combining LPCC and NN, and 87.55%—for the case of combining MFCC and GMM [10]. Yusnita et al. [20] used a feedforward multilayer perceptron (FF-MLP) consisting of two layers as a classifier, and achieved a maximum recognition accuracy 99.01% while classifying among three accents using their own dataset of audio recordings. As input features, the statistical parameters of mel-spectrograms were used.

In ASR, human speech can be described by various phonetic and prosodic features that affect the perception of accent to varying degrees. Speech is a multi-layer structure which can be analyzed at different levels, from sounds (phonemic sequences) up to melody and rhythm. Meanwhile, the physical and acoustic features of the speech signal can be considered as well. As we can see from the recent research works in Section 2, using MFCC as input characteristics is one of the most common approaches in ASR solutions [5,10,13,17]. Specifically, in [5] the accuracy of accent classification was evaluated with the use of MFCC, spectrogram, chromogram, spectral centroid, and spectral rolloff as input features. The authors of [5] also suggested that further experiments are required to check the promising case of combining MFCC with other types of available characteristics.

In this work, we test this hypothesis using MFCC in combination with other spectral characteristics. Specifically, we investigated whether using time–frequency and energy features (as recommended in [30]) could improve the automatic accent detection accuracy when used jointly with MFCC as input features. We describe the experimental environment and results, demonstrating that the greatest contribution to recognition is made by the presence of stable time–frequency patterns of energy distribution, represented by amplitude mel-spectrograms on a linear scale, which alone could be fed into the classification model as mel-spectrogram captures all of the relevant pronunciation-specific details [31].

3. Materials, Methods and Tools

A standard approach used in ASR assumes that the CNN works with the inputs which are in fact two-dimensional images representing the audio signal features [5]; thus, the number of neurons at the CNN input layer is equal to the number of characteristics of each feature vector [10]. The output value of the accent detection classifier is the probability distribution vector which attributes the speech sample to a specific accent class (where the classes correspond to the languages).

3.1. Adopting the CNN Model to Speech Signal Processing

Sound waves are complex non-stationary signals, which explains why the direct classification of sound recordings is rarely used. Selecting and extracting the best representation of an acoustic signal is an important task in ASR design, since this decision significantly affects the recognition quality and efficiency.

Accents can be understood as a composition of the phonemic and prosodic components of pronunciation: sounds, linking, intonation, stress, rhythm, etc. Accent-sensitive information could be obtained from the signal directly, but the more conventional way is when a raw audio signal undergoes a specific time–frequency transformation to calculate more sophisticated speech parameters, e.g., MFCC [5]. After the features are extracted, machine learning methods perform accent classification [10,13,30]. Our methodology is largely defined by the decisions which should be made in the course of all stages of automatic accent detection and considers four main topics—dataset, feature selection, batch normalization and machine learning model design.

3.2. Data Collection

All the experiments were made with speech samples from the **Speech Accent Archive** [29] maintained by George Mason University. The **Speech Accent Archive** is a crowd-sourced collection of speech recordings of the following text passage:

> *"Please call Stella. Ask her to bring these things with her from the store: Six spoons of fresh snow peas, five thick slabs of blue cheese, and maybe a snack for her brother Bob. We also need a small plastic snake and a big toy frog for the kids. She can scoop these things into three red bags, and we will go meet her Wednesday at the train station."*

The archive contains meta-information about the demographic and linguistic background of speakers. In total, the archive contains 2982 samples (as of the last known update at https://accent.gmu.edu/ accessed on 1 July 2022) recorded in more than 200 different native languages. The accents were classified based on the native language of the speaker.

It is important to mention that the dataset from the **Speech Accent Archive** conforms to the major ASR suitability requirements, namely:

- **Speaker diversity** assuring an adequate representation of different varieties of pronunciation;
- **Uniformity of material** referring to the same content and context;
- **Phonetic balance** when individual phonemes do not occur too often;
- **Presence of a semantic load of sentences** avoiding semantic factors that might affect pronunciation [17];
- **Working with speech segments** rather than independent words.

The latter aspect is extremely important since pronunciation patterns for words spoken separately differ from phrasal patterns expressed in the the context of related speech because of eventual assimilation (in which phonemes become similar to neighboring phonemes) or elision (where phonemes are omitted).

3.2.1. Data Classes (L1 Languages)

We used a subset of 9 language groups. These groups were labeled according to L1 as Germanic languages (English (EN), German (GE), Dutch (DU), Swedish (SW)), Romance languages (Spanish (SP), Italian (IT), French (FR)) and Slavic languages (Polish (PO), Russian (RU)). The distribution of available recordings during the experimental period according to L1 classes was as shown in Figure 1.

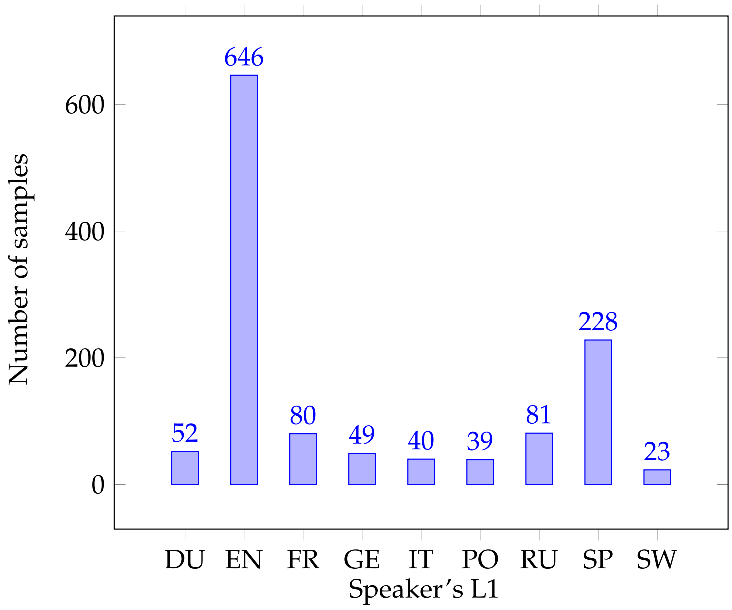


Figure 1. Distribution of audio recordings by classes (during experimental period).

Uneven distribution of recordings belonging to the different classes might deteriorate the accuracy of recognition for classes represented with fewer examples, thus, worsening the quality of the classification in general. On the other hand, using all the available examples belonging to the classes containing a much larger number of recordings might lead to heavier computations without significant improvements in recognition accuracy. Therefore, for larger groups, we limited the number of used samples by 80 recordings.

3.2.2. Preparing Audio Files for Recognition

The problem of speech signal recognition differs from the recognition of static images. In speech recognition, the object of analysis is the dynamic process and not a static image or pattern. Thus, a recognizable speech pattern is represented by feature vectors rather than a single vector. Since the presence of an accent is affected by many factors, people can have a hybridization of accents. Recognizing the accent at any point in time may be a better solution than over the entire audio signal; that is why signal segmentation is used. According to [5], classifying short segments of an audio file will more accurately classify the speaker's accent.

Thus, audio recordings with a sampling rate of 22,050 Hz were split into multiple consecutive frames of 25 ms, each with an overlap of 10 ms based on the experimental investigations discussed in Section 4.2.1.

The downside of using crowd-sourced datasets is that neither the recording environment nor the recording equipment is consistent between speakers, resulting in significant sample noise and differences in recording volume [17]. Therefore, in order to reduce the differences between audio recordings in the form of linear distortions, before training the model, the obtained data needed to be normalized within each audio recording, for example, using z-normalization (using z-score):

$$x' = \frac{(x - \mu)}{\sigma}, \tag{1}$$

where μ—mean value, σ—standard deviation.

3.2.3. Fragments of Silence

Table 2 reports our experiments on how the presence or absence of pauses in the audio files affects the classification results. There is another important but contentious aspect, namely, whether one needs to keep or remove the fragments of silence (pauses) from the input to achieve the best recognition quality. To resolve this dilemma, we arranged experiments for both approaches. For these preliminary investigations, we used a restricted set of characteristics including 13 MFCCs and fundamental frequency F_0 only.

Table 2. Comparing classifiers with preserved or removed fragments of silence.

	Fragments of Silence			
L1	**Preserved**		**Removed**	
	Accuracy	**Error**	**Accuracy**	**Error**
EN RU SP SW	0.71	0.83	0.70	0.84
FR IT SP	0.71	0.73	0.68	0.82

As we can see from Table 2, the presence of pauses can be a strong indicator of a foreign accent. For Romance languages, the difference is more noticeable than for Germanic ones. Therefore, we decided to keep the fragments of silence in audio recordings for the further accent classification experiments. Thus, the audio files processed in all the subsequent experiments were used in their original form, i.e., with all the pauses preserved.

3.3. Feature Selection

A common approach to speech signal processing is to use short-term analysis, assuming that the signal characteristics remain unchanged within a short time frame. Thus, speech utterances were compared to the feature vectors, presumably differing in their distribution with different speakers' L1s. For speech signal analysis, the frame length was near 10–30 ms, with an overlap between the frames equal to approximately half their length [10]. The signal was split into 25 s fragments overlapping by 10 s.

Audio signal characteristics were extracted from the frames using an applicable feature extraction method, such as constructing a compact representation of an audio signal using a set of mel-frequency cepstral coefficients (MFCC), resulting from a cosine transformation of the real logarithm of the short-term spectrum represented on a mel-frequency scale [30]. The latter is arguably based on the studies of the ability of the human ear to perceive sounds at different frequencies [10].

MFCC uses a spectrum capable reflecting a phoneme utterance, representing a curve in the amplitude–frequency plane, which makes it applicable to speech recognition tasks [5]. To find MFCCs, the signal was divided into short frames, then a window function was used. Discrete Fourier transform was performed giving a periodogram of the original signal. Filters were applied to the periodogram, evenly spaced on the mel-axis, which yielded the output in the form of a spectrogram. The spectrograms were then represented on a linear or logarithmic scale. The last step in finding the MFCCs was to apply the discrete cosine transform to decorrelate the resulting coefficients. Since the human ear perceives a limited range of frequencies, ASR problems usually use the first few MFCCs as input features, often limited to 13 MFCCs [5,17]. In our work, we used the same number of MFCCs (13).

Our feature set was formed on the base of **amplitude mel-spectrograms on a linear scale**. The audio signal frequencies f were converted to mel-spectrograms $M(f)$, as follows:

$$M(f) = 2595\log_{10}(1+\frac{f}{700}). \tag{2}$$

Compared to logarithmic amplitude mel-spectrograms, power mel-spectrograms, and SFF mel-spectrograms, linear amplitude mel-spectrograms performed better at classifying accents. We experimented with mel-spectrograms with 32, 64 and 128 bands as input features and discovered that the optimal balance between learning rate and recognition accuracy can be achieved using mel-spectrograms with 64 bands (see Section 4 for details).

As suggested in [5], combining MFCC with additional features can contribute to further improvements in recognition accuracy. In our work, we arranged a number of representative experiments to verify this hypothesis. We experimented with six additional features used to extend the MFCC-based model:

- **Spectral centroid** (SC) represents "center of mass" of the input sound, which formally corresponds to the frequency at which the energy of the spectrum is concentrated:

  $$C_t = \frac{\sum_{n=1}^{N} M_t[n] * n}{\sum_{n=1}^{N} M_t[n]}, \tag{3}$$

  $M_t[n]$ being the value of the frame signal spectrum t of the frequency interval n, Hz.
- **Spectral rolloff** (SR) is a measure of the asymmetry of the spectral shape of the signal. It represents the frequency R_t, such as a given percentage (usually 85%) of the total energy of the spectrum that lies below R_t. In order to calculate this value, one needs to find the proportion of frames in the signal power spectrum, where a given percentage of power falls on lower frequencies. Thus, the spectral rolloff is a frequency R_t such as:

  $$\sum_{n=1}^{R_t} M_t[n] = 0.85 * \sum_{n=1}^{N} M_t[n], \tag{4}$$

  where $M_t[n]$ is the value of the frame signal spectrum t of the frequency interval n, Hz. This value is used to determine vocalized sounds in speech, since the unvoiced sounds have a large proportion of the energy contained in the high frequency range of the spectrum.
- **Chromagram** is usually a 12-dimensional feature vector representing the amount of energy for each of the signal's height classes (such as C, C#, D, D#, E, etc.).
- **Zero Crossing** (ZCR) represents the number of signal sign changes within a segment. The ZCR feature can be helpful in describing the signal noisiness:

  $$ZCR = \frac{1}{T-1}\sum_{t=1}^{T-1} II(\{S_{t-1} < 0\}, \tag{5}$$

where S_t is a signal of duration t, $II\{X\}$ is a characteristic function whose value is equal to 1 if condition X is satisfied and 0, otherwise. For unvoiced speech, the ZCR characteristic takes on higher values.

- **Root mean square** (RMS) is a standard measure representing the average signal strength:

$$x_{RMS} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1}|x[n]|^2}. \tag{6}$$

Calculating RMS directly from the audio recordings is faster because it does not require calculating STFT. However, using a spectrogram can give a more accurate representation of signal energy over time because its frames can be split into windows. Since the characteristics of the signal can be stored in an external file in advance (before training the model), decreasing the extraction time was not critical. That is why, in our case, to improve the signal representation accuracy, RMS was calculated based on the signal spectrogram.

- **Fundamental frequency** (F_0) is the lowest frequency of the periodic signal. F_0 is the frequency at which a person's vocal cords vibrate while producing the voiced sounds. The fundamental frequency F_0 carries a lot of information about the pitch of the voice at any given time, and therefore, about the overall intonation of the speech. It has been studied that F_0 makes a significant contribution to the perception of foreign accents [6], which is especially noticeable for Germanic and Romance languages [32]. An estimation of the fundamental frequency of the signal has been carried out using the autocorrelation-based YIN algorithm [33]. According to this algorithm, initially, a cumulative mean normalized difference function is computed for short overlapped audio fragments. Then, the smallest lag giving the minimum of the normalized difference function below the threshold is chosen as the period estimate of the signal. Finally, the period estimate before converting to the corresponding frequency is refined using parabolic interpolation. Since there is no upper limit to the frequency search range for YIN, this algorithm is also suitable for higher voices. In addition, YIN is a relatively simple algorithm that can be implemented efficiently with low latency, and requires few parameters to be tuned.

To sum up, the first input feature set includes 30 audio descriptors, namely: 13 MFCCs, 12 chroma coefficients, SC, SR, ZCR, RMS and F_0.

3.4. Batch Normalization

Training a deep neural network is a complex process involving the distribution of the input data for each layer; changes in the parameters of the current layer impact subsequent layers. Thus, small changes in network parameters are amplified as the network gets deeper. This, in turn, slows down training because it requires a lower learning rate and careful parameter initialization. This phenomenon is often called "intrinsic covariant shift" [34]. In this case, covariance refers to feature values and the issue of possible internal covariant shift may be resolved through batch normalization [35]. Batch normalization can improve the performance of artificial neural networks, even in the presence of correlation between input values, while being part of the model architecture and being performed in hidden layers for each mini-batch during the training stage. The use of mini-batch is preferred over separate input values at each training step. The mini-batch error gradient is:

$$\frac{1}{m}\sum_{i=1}^{m}\frac{\partial l(x_i,\Theta)}{\partial\Theta}, \tag{7}$$

where m is the size of the mini-batch, Θ is the error minimization function, and x_i are the dataset input values, the error gradient estimate for the entire dataset.

Covariant shift poses a problem in machine learning because the learning function tries to fit the training data, and should the distribution of the test and training data differ, using the learning function may lead to erroneous results [36]. Commonly used machine learning methods work well under the assumption that the input parameters in the test and training samples belong to the same feature space and have the same distribution. In this case, when the distribution changed, the underlying statistical models needed to be rebuilt from scratch using the new training data [37].

3.5. Classification Model

The classification model for accent detection was built on CNNs used in [5]. The model consisted of two convolution layers with ReLU activation function $ReLU(x) = max(0, x)$, where x was the value of the output neuron and two-dimensional filters. The first and second convolution layers contained 32 and 64 blocks, respectively. After each convolution layer, batch normalization and pooling were applied. The flat layer was followed by two dense layers of direct propagation.

The first dense layer consisted of 128 neurons and had the ReLU activation function. For the second layer, we set the number of neurons equal to the number of accents and used the softmax activation function:

$$softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}, \tag{8}$$

where z_i is an element of the input vector of real numbers z, C is the number of classes.

The input of the model was a feature matrix extracted from audio signals.

Following the approach in [5], for the basic implementation of the model, the convolution filters with size (3, 3) and pooling layers (2, 2) with a stride of 2 were selected. To prevent overfitting, we used the dropout method with a variable probability of any neuron turning to zero—depending on the type of input data, a value from 10 to 50% was used. We used categorical cross-entropy as a loss function during training:

$$E = -\sum_{i=0}^{N}\sum_{c=0}^{C} \hat{y_i}^{(c)} \log_{10}(y_i^{(c)}), \tag{9}$$

C is the number of classes, N is the number of elements in the dataset, $\hat{y}_i \in \mathbb{R}^{10}$ is the expected probability distribution represented in vector form one-hot encode.

The earning loss function is minimized using the adaptive moment estimation (Adam) algorithm [38], where the constant learning rate coefficient is 0.001, and the parameters β_1 and β_2 are 0.9 and 0.999, respectively.

The test data were about 25% in the case of using mel-spectrograms as input data, and 15% in other cases. Figure 2 draws the workflow.

As the tools for CNN modeling, training, implementation and visualization, we used a number of standard Python libraries. In particular, the accent classifier was implemented and trained using the Keras library, providing a high-level interface to the Tensorflow computing platform. Librosa digital signal processing library was used for audio signal processing and extraction of the input characteristics. The classification quality metrics were calculated using the Scikit-learn package. Matplotlib library was used to visualize the results of the experiments. The Comet.ml platform (https://www.comet.ml/ accessed on 8 August 2022) was used to present the results of network training, to build error matrices (confusion matrices), as well as to save the statistics (the results obtained, the source code, the set of hyperparameters used, the graphs plotted, etc.) on a remote server.

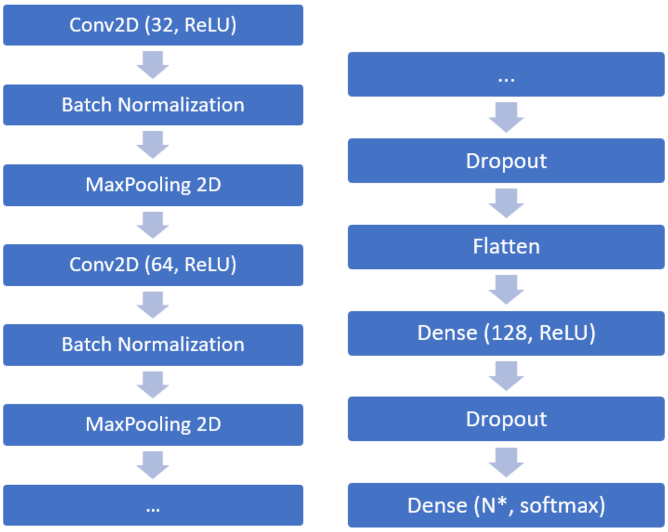


Figure 2. Classification process model. * *N*—number of recognition classes.

4. Experiments and Results

The first part of our experiments considered the architecture of CNN across hyperparameter selection, regularization and data augmentation. The second part was about bringing together various acoustic features fed to the input layer of the CNN model to improve accent recognition accuracy. All the experiments were performed for several classes of European accents.

4.1. CNN Model Tuning and Data Augmentation

The kernels of CNN convolution layers are, in fact, convolution filters, where the cross-correlation operation takes place. Kernel size refers to the width and height of the filter mask. The most common filter sizes for convolution layers in machine learning problems are (3, 3) and (5, 5).

Better overall training results often stem from using smaller filters, which require less computational power and fewer backpropagation weights. However, it is important to note that no single value is suitable for all models: filter sizes need to be optimized based on the particular type of task.

Since neighboring pixels are highly correlated, pooling can be used to reduce the size of the output data. The farther two pixels are from each other, the less correlated they are expected to be. Thus, a larger step in the pooling layer leads to more information loss. The standard pooling stride is (2, 2).

Different filter size configurations are used for different kinds of input features. The basic model of the classifier uses the filters of size (3, 3) in convolution layers and (2, 2) in pooling layers. Following the recommendations from [5], for a set of 30 characteristics (described in Section 3.3, except for amplitude of mel-spectrograms), we used 2D filter configurations for convolutional layers, as summarized in Table 3.

Table 3. Using different filter sizes for MFCC with additional features.

L1	Most Effective Settings		Error Compared to Kernel Size = (3, 3) and Pool Size = (2, 2)	
	Kernel Size	Pool Size	MFCC	30 Attribute
PO RU	(3, 3)	(3, 3)	−6%	−4%
FR IT SP	(5, 5)	(3, 3)	−8%	−11%
DU EN GE SW	(3, 3)	(3, 3)	+3%	−13%
EN RU SP SW	(3, 3)	(3, 3)	−11%	−16%
EN GE IT PO	(3, 3)	(3, 3)	−4%	−9%
DU EN FR RU	(5, 5)	(3, 3)	−5%	-

Using linear amplitude mel-spectrograms as the input for classifying among {FR, IT, SP} accents, a number of filter configurations were tried. The length of the input feature matrices used to represent the input data was 100. The learning process was stopped as soon as the change in the recognition accuracy was less than 1% within 10 epochs. The highest recognition accuracy and a relatively short model training time were achieved when using the filters of size (3, 3), namely, 99.04%, both in convolution layers and in pooling layers, as Table 4 shows.

Table 4. Results of using different filter sizes with mel-spectrograms.

French, Italian, Spanish (Romance Languages)				
Kernel Size	**Pool Size**	**Learning Time (mm:ss)**	**Accuracy**	**Error**
(3, 3)	(2, 2)	41:06	0.9889	0.0614
(3, 3)	(3, 3)	20:01	0.9904	0.0261
(5, 5)	(3, 3)	17:57	0.9852	0.0564
(7, 7)	(3, 3)	26:14	0.9867	0.0468

Thus, filters of size (3, 3) in hidden layers are the most universal and optimal for the considered CNN within the framework of ASR. The inclusion of additional features to MFCC improves the quality of recognition for many sets of languages, which confirms the hypothesis that improvements in ASR may result from combining MFCC with other types of available characteristics.

During the data augmentation phase, we tested the cases with a maximum horizontal shift of 5% and 10% for a subset of data, including the audio recordings for the foreign accent group {RU, SP, SW} as well as the audio files without a foreign accent (EN). MFCC was used as input data, as well as their alternate combinations with fundamental frequency and spectral centroid. The results are presented in Table 5.

Table 5. Classification results at different shift percentages for a set of languages of different language groups.

English, Spanish, Swedish, Russian (Mixed Group)				
Features	**Maximum Horizontal Shift during Augmentation**			
	0.05		**0.1**	
	Accuracy	**Error**	**Accuracy**	**Error**
MFCC	0.65	0.94	0.65	0.86
MFCC + F_0	0.68	0.84	0.72	0.78
MFCC + spectral centroid	0.65	0.93	0.66	0.86

Based on these results, we hypothesized that increasing the percentage of data shift may lead to higher recognition results. Given this assumption, we trained the classifier on the data set of accents {FR, IT, SP}, to which augmentation was applied. The result in Table 6 led us to conclude that the optimal accuracy/error value was reached when the maximum percentage of horizontal shift during data augmentation is about 20%.

Table 6. Classification results at different shift percentages for a set of Romance languages.

French, Italian, Spanish (Romance Languages)		
Horizontal Shift	**Accuracy**	**Error**
0.05	0.75	0.62
0.1	0.75	0.59
0.15	0.74	0.62
0.2	0.77	0.55
0.25	0.75	0.58
0.3	0.76	0.58

4.2. Input Acoustic Feature Sets

Acoustic feature sets fed into the CNN model were examined from three vantage points to obtain the feature set which yields the best recognition accuracy: input data dimensionality, possible MFCC combinations with other acoustic features and the impact of mel-spectrograms, which turned out to be the most accent-dependent, and thus, the most eloquent input feature to improve classifier performance.

4.2.1. Dimension of Input

While working with speech signals, it is necessary to consider the patterns of change in the characteristics describing these signals over time, since speech is viewed as a time-dependent function. Thus, it is essential to consider the sequences of feature vectors rather than a single one-dimensional vector.

The division of the input features into larger or smaller chunks may introduce bias. Larger chunks can enable discovering longer speech patterns (more likely to be accent-dependent), but the training set becomes smaller, and training on high-dimensional data is naturally more computationally expensive (and therefore, slower). Selecting shorter fragments allows using more input data, but can deteriorate the information captured about the accent from a fragment of feature vectors.

To find the optimal size of the input feature matrices, a series of experiments were arranged in which the feature vectors were grouped into blocks of size ranging between 30 and 500 vectors per block. Tables 7 and 8 list the results.

Table 7. Classification results with different sizes of input matrices for Slavic and Romance languages (MFCC).

Size of Feature Matrices	**Accuracy**	**Error**
Russian, Polish (Slavic Languages)		
30	0.8075	0.4248
50	0.8315	0.3946
70	0.8151	0.4458
100	0.835	0.3828
150	0.8537	0.3573
200	0.8156	0.3956
300	0.8742	0.372
500	0.8065	0.4547
French, Italian, Spanish (Romance Languages)		
30	0.6337	0.7862
50	0.6804	0.7346
70	0.6696	0.7371
100	0.7088	0.6652
150	0.7393	0.6238
200	0.7318	0.6699
300	0.7548	0.5748
500	0.7764	0.5607

The experiments were performed using the sequences of vectors of mel-cepstral coefficients as input features. The training stopped when the change in accuracy was at least 0.5% for an interval of 20 epochs or when 300 epochs were reached among five accents, and 170 epochs in other cases. The probability of a neuron reaching zero when using the thinning method was 50%.

Table 8. Classification results for different sizes of input matrices for Germanic and mixed languages (MFCC).

Size of Feature Matrices	Accuracy	Error
English, German, Dutch, Swedish (Anglo-Saxon group)		
30	0.6278	0.918
50	0.6646	0.8325
70	0.6473	0.8631
100	0.7012	0.7572
150	0.7101	0.8038
200	0.727	0.7033
300	0.735	0.7251
500	0.6866	0.8178
English, Spanish, German, Russian, French (Mixed group)		
30	0.4674	1.3059
50	0.5162	1.2053
70	0.5409	1.1692
100	0.554	1.1555
150	0.583	1.1013
200	0.5874	1.1133
300	0.6095	1.0474
500	0.4302	1.6158

Tables 7 and 8 show that by modifying the dimension of input features and the maximum percentage of horizontal image shift during data augmentation, it is possible to increase classification accuracy by about 7% compared to [5] (60.95% against 53.92%) for recognition among five accents. For classification among three accents, the highest accuracy achieved is 77.64% against 70.38% reported in [5], and against about 61% reported in [11]. Figures 3–5 illustrate how the error and accuracy value change in the process of training the classifier to distinguish among five classes.

The graphs show how an increase in the length of the input matrices to a certain value leads to a decrease in the error value when testing the model. However, the change in accuracy and error graphs becomes noisier due to a decreasing amount of input data. As the number of input instances becomes too small, the classifier fails to sufficiently capture accent-dependent patterns in speech.

From our experiments, we can conclude that increasing the size of input data blocks to a certain value leads to an improvement in recognition accuracy, which can be seen in Figures 6 and 7. However, increasing the size of the matrices is inversely proportional to the number of input instances, which leads to the inability of the model to fully capture accent-dependent patterns. For the Romance languages, the recognition accuracy increased by increasing the number of input characteristics up to 300 per block. The noise, however, also increased during the training with the extended matrices. For training the classification of Slavic accents, the maximum accuracy was achieved with the length of input data blocks equal to 150 feature vectors, 200 vectors for Germanic languages, and 300 vectors for the mixed-language group.

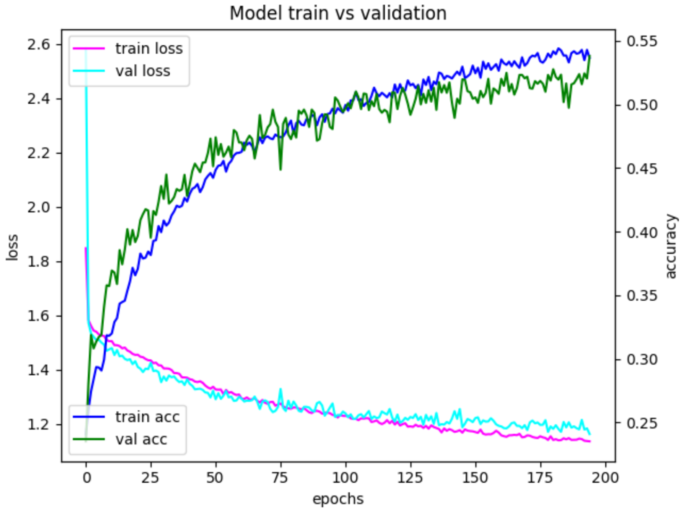


Figure 3. Variation of accuracy and error during classifier training: MFCC matrices with size 50 (EN, FR, GE, RU, SP).

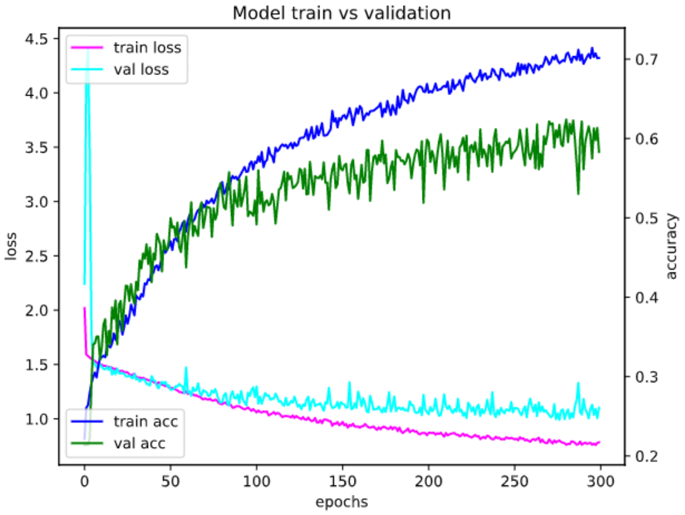


Figure 4. Variation of accuracy and error during classifier training: MFCC matrices with size 150 (EN, FR, GE, RU, SP).

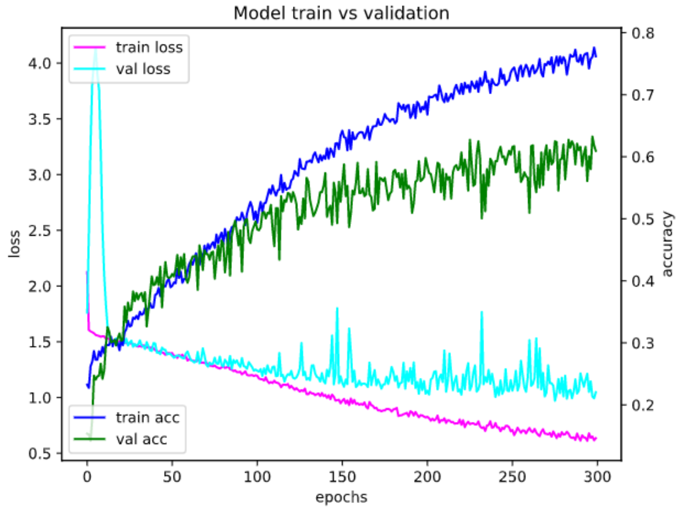


Figure 5. Variation of accuracy and error during classifier training: MFCC matrices with size 300 (EN, FR, GE, RU, SP).

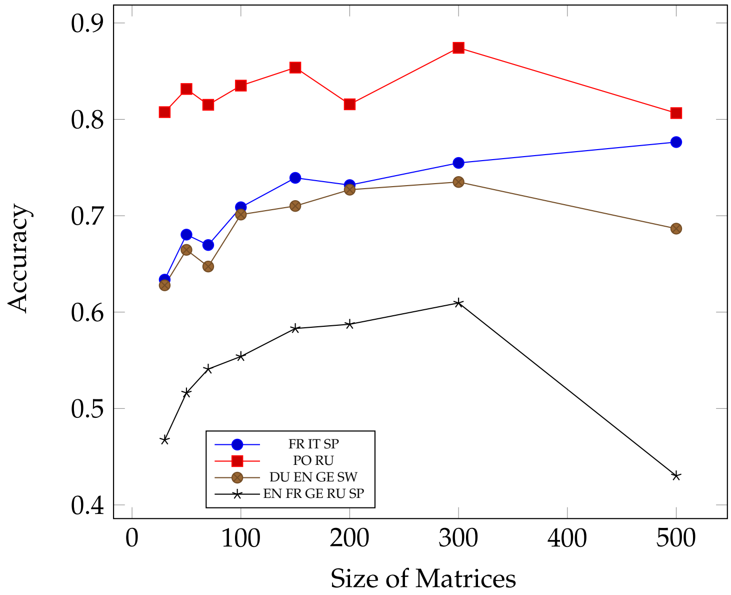


Figure 6. Accuracy variation when changing the size of the input matrices of features.

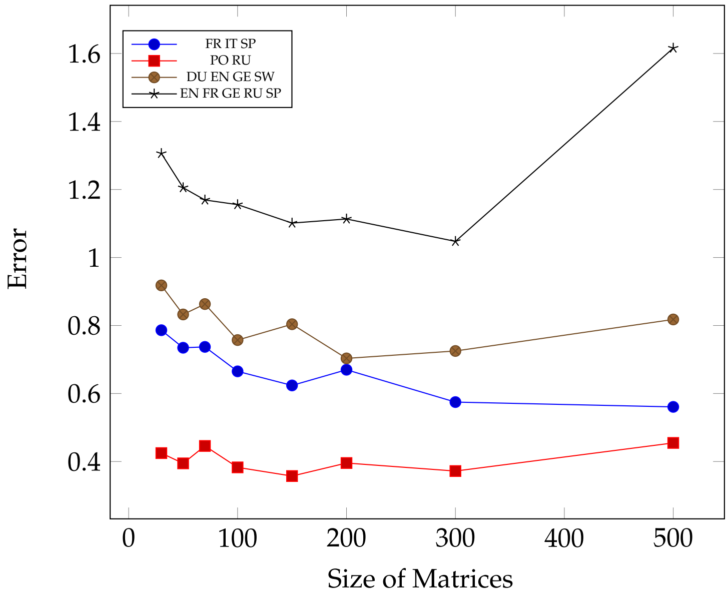


Figure 7. Error variation when changing the size of the input feature matrices.

Table 9 shows the results of an experiment performed on Romance accents using mel-spectrograms as input, varying the number of mel-bands used to represent the spectrograms.

During the experiments, we used a dropout of 0.25. The size of the filters in the convolution layers was (5, 5). The size in the pooling layers was (3, 3). The training was stopped when the recognition accuracy ceased to change by at least 1% for ten epochs.

As can be seen in Table 9, mel-spectrograms, consisting of 64 frequency bands, proved to be the most effective, and were chosen as input characteristics for recognition. Although the use of 128-band mel-spectrograms can slightly increase the recognition accuracy, it substantially increases the training time. Contrariwise, using mel-spectrograms consisting of 32 mel-frequency bands is less computationally expensive but leads to a significant increase in error.

Based on the experimental results summarized in Figures 3–5 and Table 9, we can conclude that the optimal size of the input feature matrices is 75 vectors when using amplitude mel-spectrograms on a linear scale.

Table 9. Classification results for different sizes of input matrices for a set of Romance languages (mel-spectrograms).

French, Italian, Spanish (Romance Languages)				
Number of Mel-Bands	**Size of Input Matrices**	**Training Time (hh:mm:ss)**	**Accuracy**	**Error**
32	30	00:11:06	0.8889	0.2837
	50	00:13:10	0.9239	0.2004
	75	00:13:20	0.9586	0.1216
	100	00:14:53	0.9483	0.1539
	150	00:12:16	0.9207	0.2347
	200	00:15:03	0.8675	0.3186
64	30	00:28:55	0.9737	0.0858
	50	00:19:48	0.9987	0.0406
	75	00:19:16	0.9912	0.033
	100	00:17:57	0.9852	0.0564
	150	00:25:15	0.9877	0.0398
	200	00:36:25	0.9593	0.1443
128	100	01:23:56	0.9985	0.0074
	150	01:31:14	0.9832	0.0873

4.2.2. MFCC Combined with Additional Features

Now, let us consider the case of extending MFCC with a number of additional features, as suggested in [5]. MFCC speech characteristics are widely used in accent detection because they provide a compact yet informationally dense representation of an audio signal, resulting in high-classification accuracy. In [5,30], it was suggested that the accuracy can be further improved by adding additional information to MFCC. However, adding an arbitrarily large number of input features would be detrimental, since excessive information would slow down the classifier's training process and increase the model overfit due to the noise. Therefore, it is important to select a limited number of suitable representative characteristics. Hence, in this work, we strived to discover such essential characteristics for an MFCC extension that would positively affect classification accuracy while maintaining the basic filter sizes in the hidden layers of the classifier. It is worth noting that our feature selection does not contradict to Fisher criterion [39–41], though, in our work, we did not use it explicitly.

The training was stopped when the training accuracy of 90% or 120 epochs was reached for all accent sets, except for the case {EN, RU, SP, SW}, where the training process terminated as soon as 350 epochs were reached. The results obtained are shown in Tables 10 and 11.

Obtained values of the testing error demonstrate that in half of the cases with filter sizes $(3, 3)$ in convolutional layers and $(2, 2)$ in pooling layers, the accent-dependent patterns captured are worse compared to only using MFCC as input characteristics of audio signals.

In the case of the accent group {EN, GE, IT, PO}, adding the fundamental frequency to the mel-cepstral coefficients helped to increase the recognition accuracy by approximately 3%. For the set {EN, RU, SP, SW}, the most effective selection was to use all types of additional characteristics. The increase in classification accuracy was approximately 3% compared to the usage of MFCC alone.

Intonation makes a significant contribution to the recognition of a foreign accent. Based on the fact that the F_0 contour in most experiments did not improve the classification results, we can conclude that a description of intonation was contained within MFCC. When extracting MFCC, information about F_0 was partially preserved due to the close distance between the low-frequency channels of the mel-filters [42].

Table 10. Classification results using different types of input features for Slavic and Romance accents.

Features	Test Accuracy	Test Error
	Russian, Polish (Slavic Languages) Threshold Accuracy—0.72	
MFCC	0.84	0.37
MFCC + F_0	0.83	0.4
MFCC + spectral centroid	0.85	0.39
MFCC + spectral decay	0.84	0.4
MFCC + chromogram	0.79	0.44
MFCC + ZCR	0.84	0.38
MFCC + RMS	0.83	0.41
All	0.81	0.41
	French, Italian, Spanish (Romance Languages) Threshold Accuracy—0.43	
MFCC	0.75	0.6
MFCC + F_0	0.69	0.71
MFCC + spectral centroid	0.67	0.73
MFCC + spectral decay	0.68	0.72
MFCC + chromogram	0.63	0.84
MFCC + ZCR	0.71	0.68
MFCC + RMS	0.7	0.7
All	0.66	0.8

Table 11. Classification results when using different types of input features for accents of mixed-language groups.

Features	Test Accuracy	Test Error
	English, Italian, German, Polish (Mixed group) Threshold Accuracy—0.29	
MFCC	0.62	1.00
MFCC + F_0	0.65	0.88
MFCC + spectral centroid	0.61	0.94
MFCC + spectral decay	0.63	0.96
MFCC + chromogram	Threshold not passed	
MFCC + ZCR	0.64	0.9
MFCC + RMS	0.64	0.91
All	0.6	0.95
	English, Spanish, Swedish, Russian (Mixed group) Threshold Accuracy—0.33	
MFCC	0.72	0.81
MFCC + F_0	0.71	0.83
MFCC + spectral centroid	0.68	0.88
MFCC + spectral decay	0.68	0.93
MFCC + chromogram	0.68	0.92
MFCC + ZCR	0.68	0.85
MFCC + RMS	0.67	0.95
All	0.75	0.7

4.2.3. Mel-Spectograms

Linear scale mel-amplitude spectrograms extracted from the audio signals can also be tried as the inputs of the classifier. At the same time, according to the previously established optimal parameters of the classifier, the filters of size (3, 3) are used in both the convolution and the pooling layers, while the size of the input feature matrices is 75 elements. The number of epochs was limited to 60, while the learning process was stopped when the change in recognition accuracy was less than 1% within ten epochs.

We applied a dropout for different sets of accents to eliminate overfitting with values ranging between 10% and 25%. Regularization is applied intermittently either to the training or to the test set, to cope with model redundancy and inability to generalize the data.

Figure 8 shows accuracy and loss in the training model for the largest set of accents used in the experiments. The graphs in dark blue and green show the accuracy of the training and test data, while the graphs in pink and light blue show the training and test data validation. The graphs for other combinations of accents can be found in the Appendix A.1.

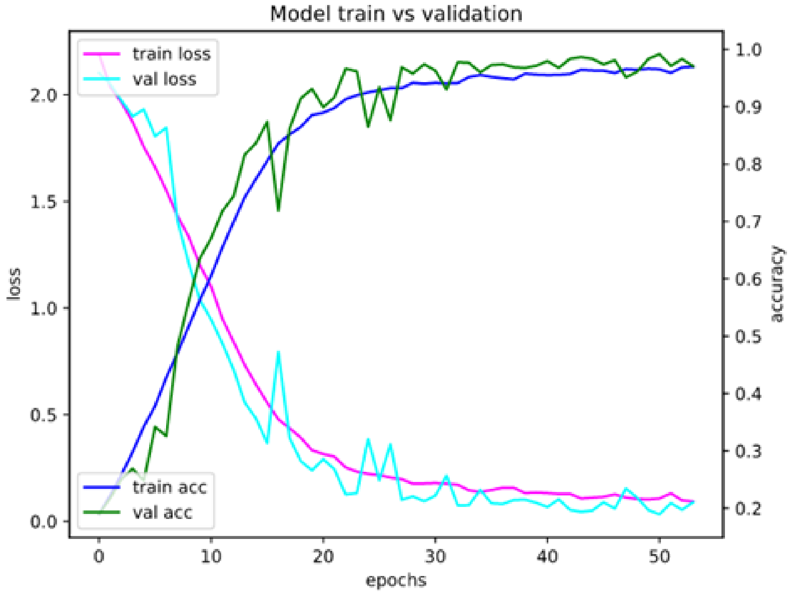


Figure 8. Accuracy and loss on training and test data during classifier training among the set of accents *DU*, *EN*, *FR*, *GE*, *IT*, *PO*, *RU*, *SP*, *SW*.

By the end of the training, the model achieved similar accuracy and loss values for the training and test data. For a smaller number of epochs compared to previous experiments, it was possible to achieve a much smaller error and greater accuracy, which means that using amplitude mel-spectrograms on a linear scale allowed the model to place broad boundaries between classes. For reference, the average number of training epochs was 46, while the process took 37.18 s, performed using a mainstream 2021 laptop computer. At the recognition stage among 9 accents, the process took 64.09 for 52 epochs using the same hardware. Accuracy and loss values achieved while testing the resulting models for classifying different sets of accents are presented in Table 12.

Table 12. Accuracy and loss for trained Accent Classification Models.

Accents	Accuracy	Loss
PO RU	0.987	0.039
FR IT SP	0.986	0.052
DU EN GE SW	0.982	0.075
EN RU SP SW	0.988	0.042
EN GE IT PO	0.985	0.053
DU EN FR RU	0.984	0.039
EN FR GE RU SP	0.978	0.071
DU EN FR GE RU SP	0.964	0.097
DU EN FR GE IT PO RU SP SW	0.986	0.044
Average	0.982	0.056

Table 13 presents the achieved results against the works reviewed in Section 2.

Table 13. Comparison of existing solutions with the obtained results.

Source	Classifier	Number of Classes Recognized	Precision	Accuracy of CNNs Trained on Mel-Amplitude Spectrograms on a Linear Scale
[17]	CNN (with attention mechanism)	2	1.0	0.987
		4	0.99	0.984
		9	0.995	0.986
[11]	CNN (AlexNet)	3	0.61	0.986
[8]	GMM	2	0.862	0.987
[10]	FFNN	6	0.914	0.964
[5]	CNN	3	0.703	0.986
		5	0.539	0.978
[20]	FF-MLP	3	0.99	0.986

Amplitude mel-spectrograms on a linear scale showed high efficiency in recognizing foreign accents in English speech. However, the results turned out to be slightly lower compared to [17]. This may be due to heterogeneous audio recordings contained in the Speech Accent Archive dataset, in contrast to the homogeneous dataset in [17], in which all entries were made using the same equipment.

Compared to other solutions using Speech Accent Archive dataset—refs. [5,11] and with [8] where the dataset was used, based on text from the Speech Accent Archive, the implemented model achieved much better recognition accuracy by tuning hyperparameters, dimensionality of input features, and selecting amplitude mel-spectrograms on a linear scale as input features. The better recognition quality compared to [8] can be explained, among other things, by the fact that the authors of [8] removed silence fragments from audio recordings before extracting characteristics. During this research, we found that pauses in speech have a positive effect on the ability to determine accents.

5. Evaluation

The quality of the CNN-based classifier can be evaluated by creating a confusion matrix and connected standard IR metrics including overall accuracy, precision, recall and $F1$.

The confusion matrix is a matrix C, where $C_{i,j}$ is equal to the number of observations that belong to class i and recognized as an object of class j. Such $C_{i,j}$, where $i = j$ is the number of observations where the object class was recognized correctly. The size of the matrix C is $N \times N$, where N is the number of classes.

Figure 9 shows the error matrix for the largest set of accents used in the experiments, while the error matrices for other combinations of accents can be found in Appendix A.2.

Based on extracting the numbers of true positive TP, true negative TN, false positive FP and false negative FN cases from the confusion matrix, the overall accuracy is as follows:

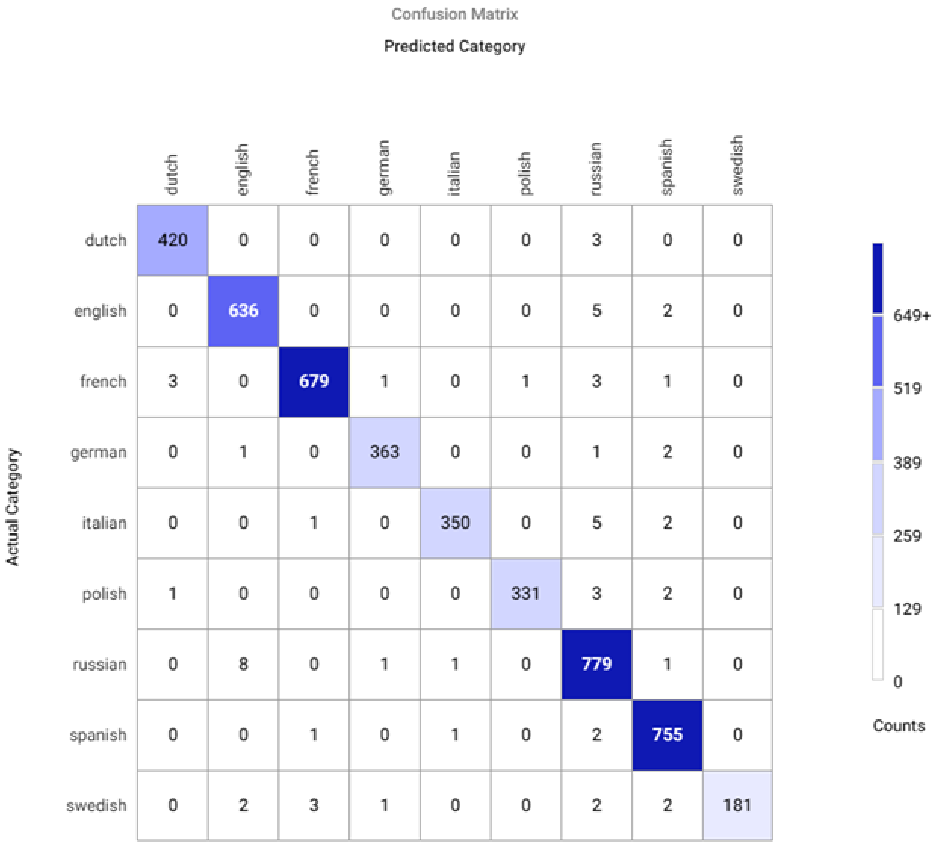
$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{\sum_{i=0}^{N} C_{ii}}{\sum_{i=0}^{N} \sum_{j=0}^{N} C_{ij}} = 0.986 \tag{10}$$

The standard precision, recall and $F1$ metrics are as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{11}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{12}$$

$$F1 = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{13}$$



	dutch	english	french	german	italian	polish	russian	spanish	swedish
dutch	420	0	0	0	0	0	3	0	0
english	0	636	0	0	0	0	5	2	0
french	3	0	679	1	0	1	3	1	0
german	0	1	0	363	0	0	1	2	0
italian	0	0	1	0	350	0	5	2	0
polish	1	0	0	0	0	331	3	2	0
russian	0	8	0	1	1	0	779	1	0
spanish	0	0	1	0	1	0	2	755	0
swedish	0	2	3	1	0	0	2	2	181

Figure 9. Classification error matrix among accents set $DU, EN, FR, GE, IT, PO, RU, SP, SW$.

Table 14 lists the computed IR metrics for each of the accent class, as well as the integral values across all the classes used in the experiments.

Table 14. Average values of precision, recall and $F1$.

Class	Precision	Recall	*F1*
DU	0.985	0.965	0.975
EN	0.984	0.983	0.983
FR	0.98	0.984	0.982
GE	0.978	0.98	0.98
IT	0.997	0.973	0.987
PO	0.993	0.977	0.987
RU	0.983	0.98	0.983
SP	0.968	0.992	0.978
SW	0.987	0.97	0.977
Average	0.984	0.978	0.981

To sum up, the average precision, recall, and F1 values for the considered accent classifier are 98.4%, 97.8%, and 98.1%, respectively. The resulting values show the good classification quality for a classifier based on amplitude mel-spectrograms on a linear scale while distinguishing among the nine classes.

6. Discussion

As a major new contribution, this work proposes an accent classification CNN model trained on amplitude mel-spectrograms on a linear scale. The model was applied to the sparse data from the Speech Accent Archive, which is a crowd-sourced collection of speech recordings. The sparsity of this dataset makes the implementation interesting with respect to real-world practical implications of potential application.

The results of the experiments with classifications of up to nine accents demonstrated that by the end of the training, the model is able to achieve similar accuracy and loss values for the training and test data. Furthermore, using amplitude mel-spectrograms on a linear scale allows the model to place broad boundaries between classes, as shown in Table 12. Linear scale amplitude mel-spectrograms gave much better accuracy and loss results than using MFCC alone or combined with additional features.

The accuracy of the model in the experiments ranged from 0.964 to 0.987 when working with nine classes of accented speech in English. A similar result using mel-spectrogram with the CNN model accent was achieved when discriminating among five accent classes of spoken Kashmiri, showing an accuracy of 0.9866 [15].

Though the techniques and features used in our work are known in the speech processing domain, extensive experiments involving their combination and the selection of optimal parameters for CNN filters have not been reported so far in their application to the specific problem of accent recognition. Compared to the reported solutions which use the same dataset, the implemented model achieved better recognition accuracy with no additional computational overhead by tuning hyperparameters and dimensionality of input features. In particular, the better recognition quality compared to [8] can be explained by using the model preserving silence fragments in the audio recording, which may correlate with the specificity of speech traits depending on the speaker's L1.

In connection to our project on developing a CAPT system involving speech prosody evaluation and modeling [43,44], this article also enhances our understanding of how intonation may impact accent recognition. Based on the fact that the F_0 contour in most experiments did not improve the classification results, we concluded that intonation features are subsumed within MFCC. When extracting MFCC, information about F_0 is partially preserved due to the close distance between the low-frequency channels of the mel-filters [42].

Accent recognition can also be one of the steps towards assessing the pronunciation of a foreign language in general; therefore, automatic foreign accent recognition systems can be used in computer-assisted pronunciation training systems. Successful accent identification can be one of features providing an opportunity for more instructive, better personalized, and customized feedback to language learners according to their manner of speaking [2,10] as speakers with the same accent have been observed to have similar trends in mispronunciation [9]. Therefore, more accurate accent classifications can help in improving the robustness of mispronunciation detection and diagnosis to mitigate the adverse effects of accent variety for the benefits of CAPT systems [3,45].

7. Conclusions

Let us summarize the major results and findings of the current study:

1. Using additional audio signal information on time–frequency and energy features (such as spectrogram, chromogram, spectral centroid, spectral rolloff, and fundamental frequency), mel-frequency cepstral coefficients (MFCC) are proven to increase the accuracy of the accent classification compared to a conventional feature set based on MFCC and raw spectrograms.
2. Amplitude mel-spectrograms on a linear scale (in contrast to logarithmic scale used in most studies) appear more powerful in the accent classification task and make it possible to produce state-of-the-art accent recognition accuracy, ranging from 0.964 to 0.987.
3. Reported accuracy has been achieved using heterogeneous sparse data from the Speech Accent Archive, unlike the best reported experiments, where the datasets

are prepared in standardized conditions using the same equipment. This outcome addresses real life situations, with varying recording environments and tools.

4. Based on our experiments, we demonstrated that the pauses in speech have a positive effect on the ability to determine accents. This is why they should not be eliminated from the input, at least with respect to the accent classification process.
5. The experiments conducted enhanced our understanding of how intonation may impact accent recognition. Based on the fact that the fundamental frequency contour in most experiments did not improve the classification results, we concluded that intonation features are subsumed within MFCC. To the best of our knowledge, the problem of accent recognition in connection to the analysis of language prosody features makes an important and additional novel contribution.

To sum up, the amplitude mel-spectrograms on a linear scale showed effectiveness in solving the problem of determining the speech accent in a foreign language using a CNN-based classifier, even when applied to sparse speech data from a crowd-sourced dataset. Let us note that for the case of a crowd-sourced dataset, reaching an accuracy—which is very close to the results of experiments with high-quality homogeneous data reported in many reviewed works such as [15–17]—can be considered as a very good achievement, particularly when it outperforms other models. Further studies may be helpful to expand the number of recognition classes, using an intermediate classifier to determine the L1 language group of the speaker before classifying a particular accent and using a dataset with a variety of spoken content.

Author Contributions: Conceptualization, N.B. and E.P.; methodology, V.M., M.L. and N.B.; software, I.L. and M.L.; validation, M.L. and V.M.; formal analysis, I.L.; data curation, M.L.; writing—original draft preparation, V.M., M.L. and E.P.; writing—review and editing, N.B., E.P. and J.B.; visualization, M.L.; supervision and project administration, E.P.; funding acquisition, E.P. and J.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was in part funded by the Japan Society for the Promotion of Science (JSPS), grant number 20K00838.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Source code in Python used for preparing and running the experiments can be found at https://github.com/MariaForester/AccentRecognition (accessed on 8 August 2022).

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

Abbreviations

The following abbreviations are used in this manuscript:

ASR	Automatic Speech Recognition
bLSTM	Bidirectional Long Short-term Model
CAPT	Computer-assisted Pronunciation Training
CG	Chromagram
CNN	Convolutional Neural Networks
FFNN	Feedforward Neural Network
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
IR	Information Retrieval
KNN	k-nearest Neighbor
LSTM	Long Short-term Model
MDPI	Multidisciplinary Digital Publishing Institute

MFCC	Mel-frequency Cepstral Coefficients
RF	Random Forest
RMS	Root Mean Square
SC	Spectral Centroid
SFF	Single Filtered Frequency
SG	Spectrogram
STFT	Short-time Fourier Transform
SVM	Support Vector Machine
SR	Spectral Rolloff
ZCR	Zero CRossing

Appendix A. Model Assessment: Experimental Results in Details

Appendix A.1. Accuracy and Loss for Different Sets of Recognized Accents

Figures A1–A8 show accuracy and lost on training and test data for different combination of accents.

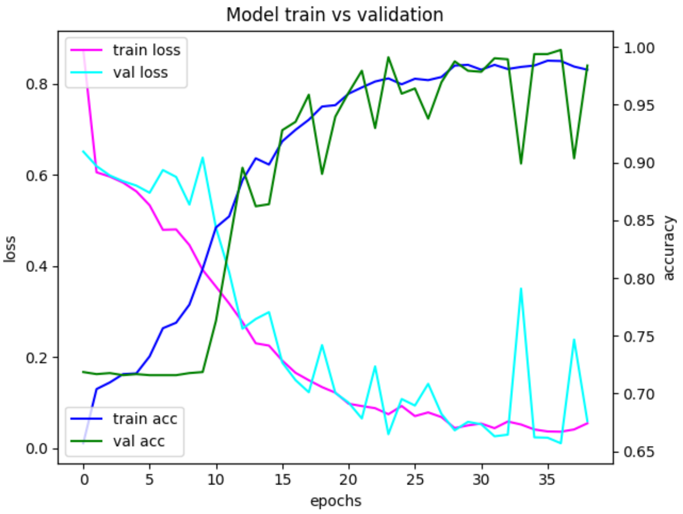


Figure A1. Accuracy and loss on training and test data during classifier training among the set of accents PO, RU.

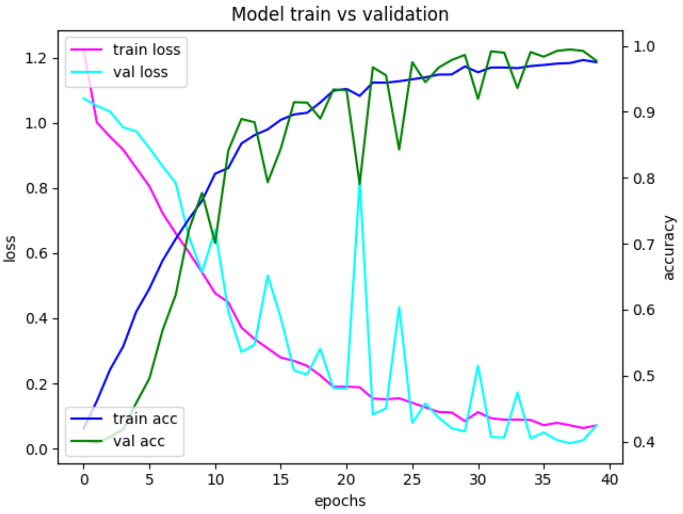


Figure A2. Accuracy and loss on training and test data during classifier training among the set of accents FR, IT, SP.

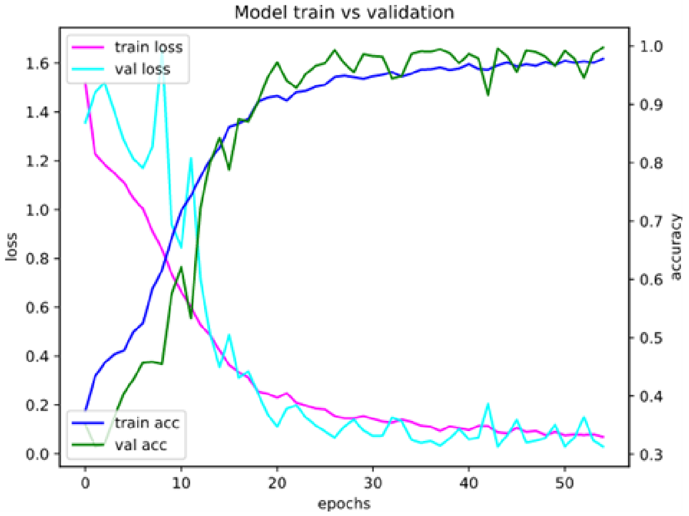


Figure A3. Accuracy and loss on training and test data during classifier training among the set of accents DU, EN, GE, SW.

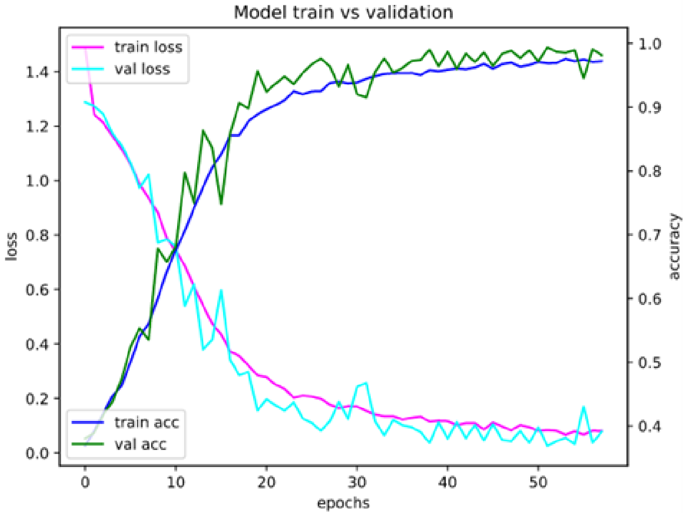


Figure A4. Accuracy and loss on training and test data during classifier training among the set of accents EN, RU, SP, SW.

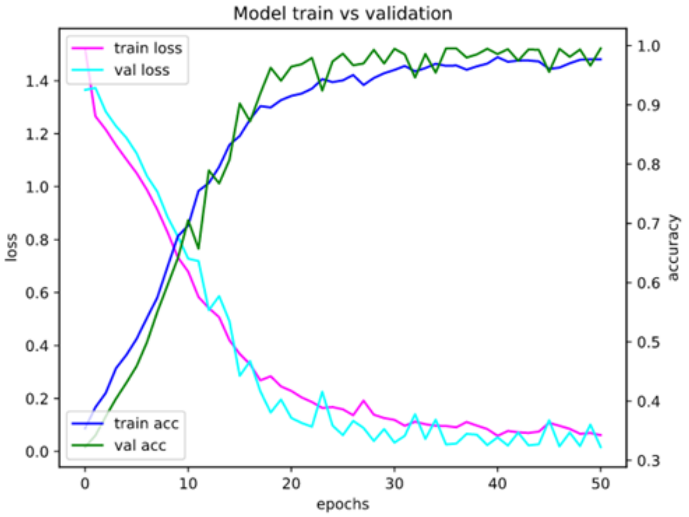


Figure A5. Accuracy and loss on training and test data during classifier training among the set of accents EN, GE, IT, PO.

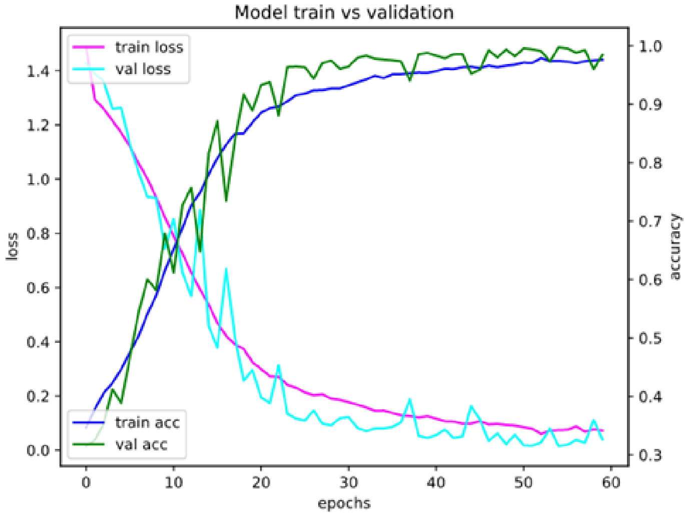


Figure A6. Accuracy and loss on training and test data during classifier training among the set of accents DU, EN, FR, RU.

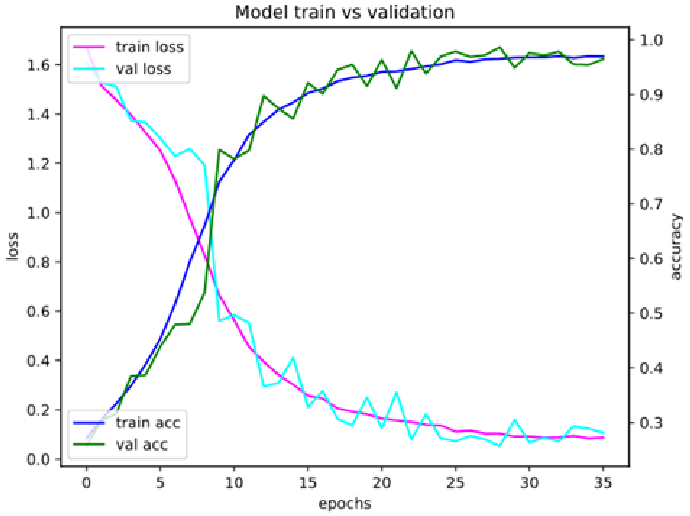


Figure A7. Accuracy and loss on training and test data during classifier training among the set of accents EN, FR, GE, RU, SP.

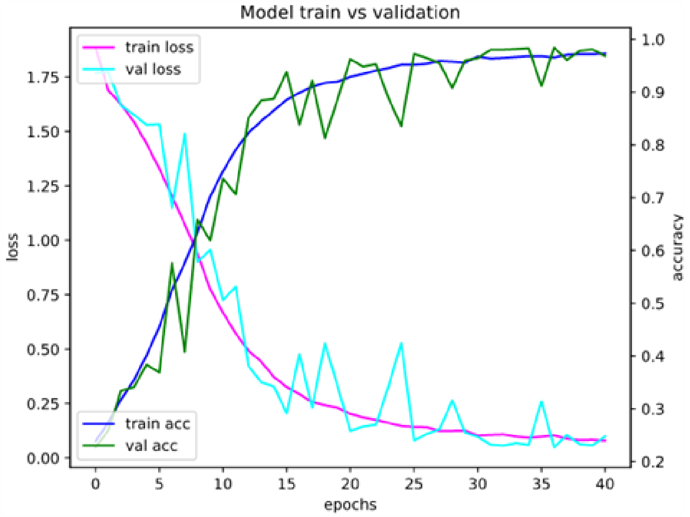


Figure A8. Accuracy and loss on training and test data during classifier training among the set of accents DU, EN, FR, GE, RU, SP.

Appendix A.2. Confusion Matrices for Different Sets of Recognized Accents

Figures A9–A16 show the error matrices obtained in the experiments corresponding to different combinations of accents.

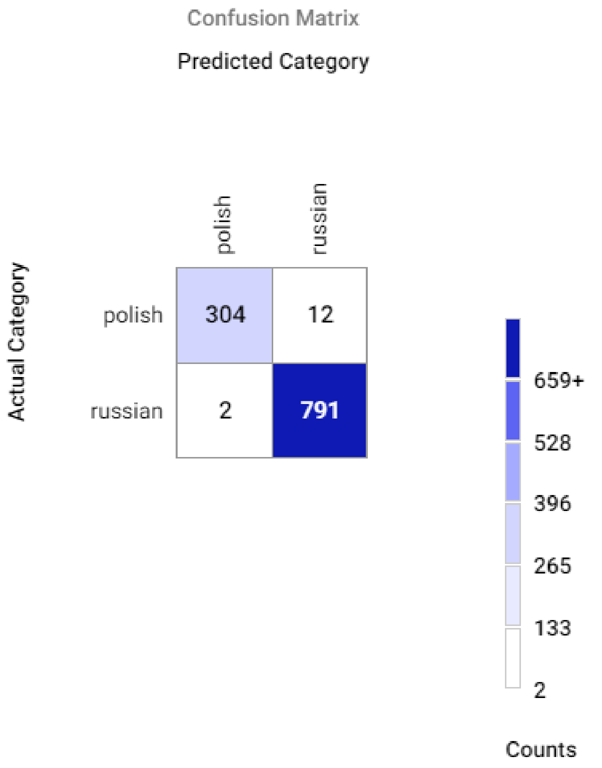


Figure A9. Classification error matrix among accents set PO, RU.

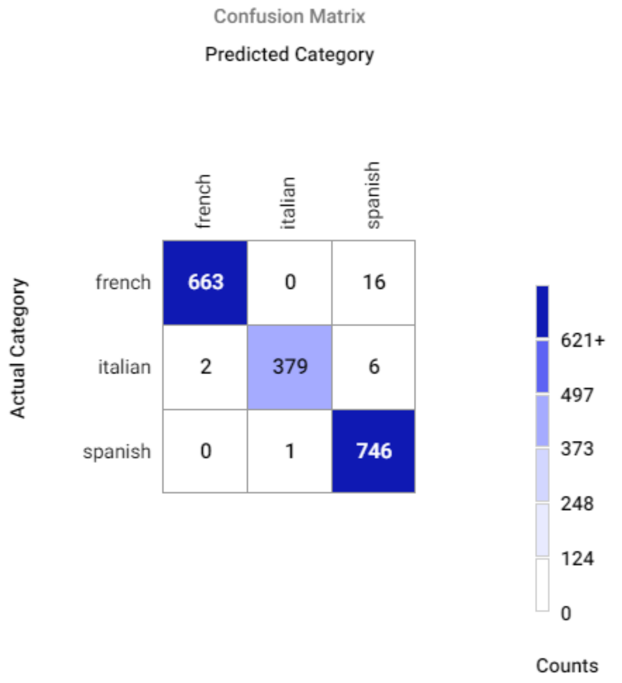
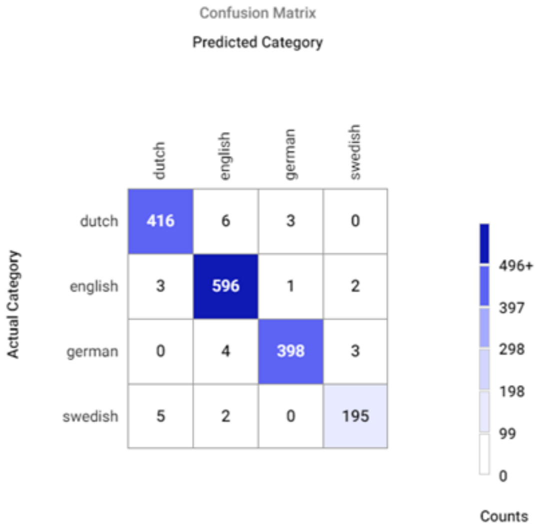
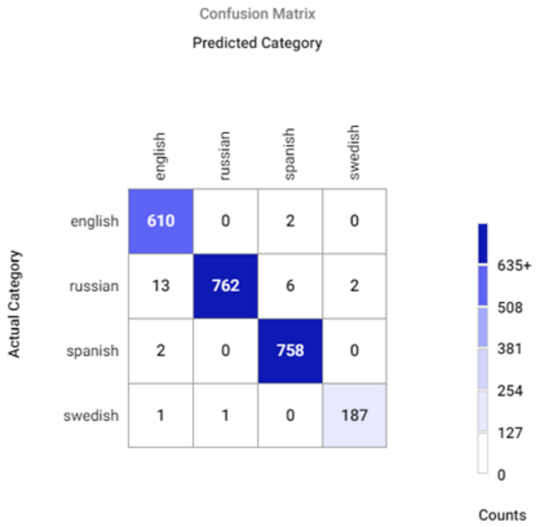


Figure A10. Classification error matrix among accents set FR, IT, SP.



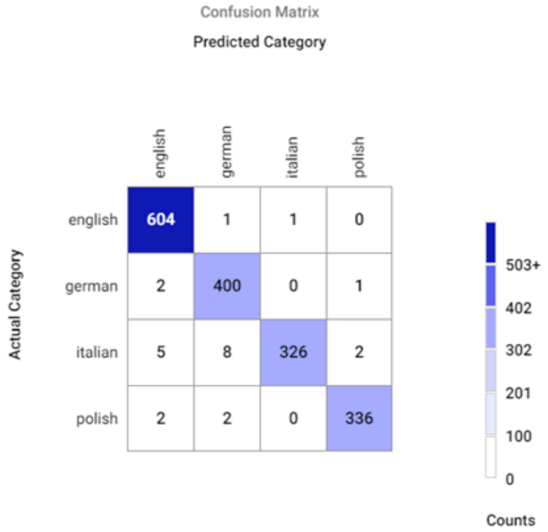
	dutch	english	german	swedish
dutch	416	6	3	0
english	3	596	1	2
german	0	4	398	3
swedish	5	2	0	195

Figure A11. Classification error matrix among accents set DU, EN, GE, SW.



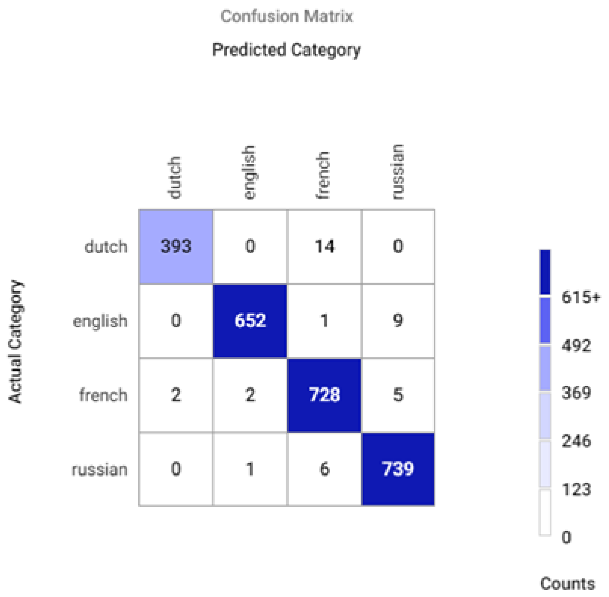
	english	russian	spanish	swedish
english	610	0	2	0
russian	13	762	6	2
spanish	2	0	758	0
swedish	1	1	0	187

Figure A12. Classification error matrix among accents set EN, RU, SP, SW.



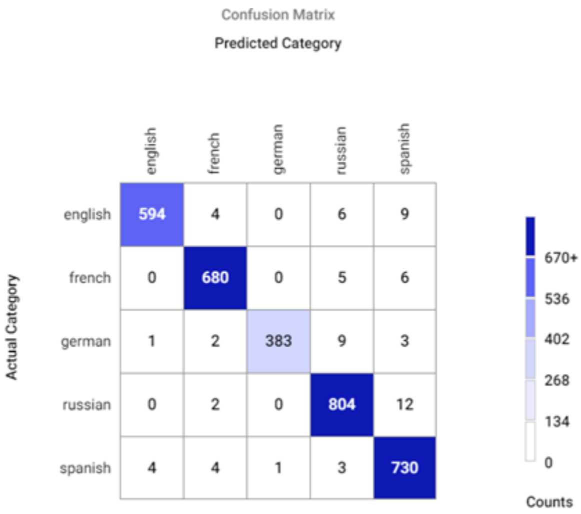
	english	german	italian	polish
english	604	1	1	0
german	2	400	0	1
italian	5	8	326	2
polish	2	2	0	336

Figure A13. Classification error matrix among accents set EN, GE, IT, PO.



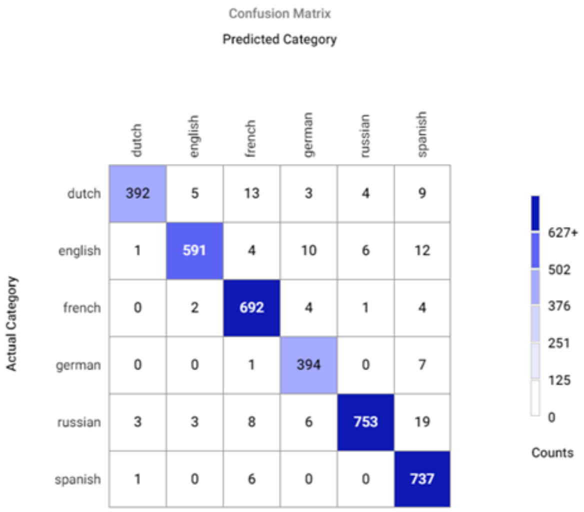
	dutch	english	french	russian
dutch	393	0	14	0
english	0	652	1	9
french	2	2	728	5
russian	0	1	6	739

Figure A14. Classification error matrix among accents set DU, EN, FR, RU.



	english	french	german	russian	spanish
english	594	4	0	6	9
french	0	680	0	5	6
german	1	2	383	9	3
russian	0	2	0	804	12
spanish	4	4	1	3	730

Figure A15. Classification error matrix among accents set EN, FR, GE, RU, SP.



	dutch	english	french	german	russian	spanish
dutch	392	5	13	3	4	9
english	1	591	4	10	6	12
french	0	2	692	4	1	4
german	0	0	1	394	0	7
russian	3	3	8	6	753	19
spanish	1	0	6	0	0	737

Figure A16. Classification error matrix among accents set DU, EN, FR, GE, RU, SP.

References

1. Boula de Mareüil, P.; Vieru, B. The Contribution of Prosody to the Perception of Foreign Accent. *Phonetica* **2006**, *63*, 247–267. https://doi.org/10.1159/000097308.
2. Rogerson-Revell, P.M. Computer-assisted pronunciation training (CAPT): Current issues and future directions. *RELC J.* **2021**, *52*, 189–205.
3. Jiang, S.W.F.; Yan, B.C.; Lo, T.H.; Chao, F.A.; Chen, B. Towards robust mispronunciation detection and diagnosis for L2 English learners with accent-modulating methods. In Proceedings of the 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Cartagena, Colombia, 13–17 December 2021; pp. 1065–1070.
4. Algabri, M.; Mathkour, H.; Alsulaiman, M.; Bencherif, M.A. Mispronunciation Detection and Diagnosis with Articulatory-Level Feedback Generation for Non-Native Arabic Speech. *Mathematics* **2022**, *10*, 2727. https://doi.org/10.3390/math10152727.
5. Singh, Y.; Pillay, A.; Jembere, E. Features of Speech Audio for Accent Recognition. In Proceedings of the 2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, 6–7 August 2020; pp. 1–6. https://doi.org/10.1109/icABCD49160.2020.9183893.
6. Hansen, J.H.L.; Arslan, L.M. Foreign accent classification using source generator based prosodic features. In Proceedings of the 1995 International Conference on Acoustics, Speech, and Signal Processing, Detroit, MI, USA, 9–12 May 1995; Volume 1, pp. 836–839. https://doi.org/10.1109/ICASSP.1995.479824.
7. Huang, H.; Xiang, X.; Yang, Y.; Ma, R.; Qian, Y. Aispeech-sjtu accent identification system for the accented english speech recognition challenge. In Proceedings of the ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 6254–6258.
8. Deshpande, S.; Chikkerur, S.; Govindaraju, V. Accent classification in speech. In Proceedings of the Fourth IEEE Workshop on Automatic Identification Advanced Technologies (AutoID'05), Buffalo, NY, USA, 17–18 October 2005; pp. 139–143. https://doi.org/10.1109/AUTOID.2005.10.
9. Huang, C.; Chen, T.; Chang, E. Accent Issues in Large Vocabulary Continuous Speech Recognition. *Int. J. Speech Technol.* **2004**, *7*, 141–153. https://doi.org/10.1023/b:ijst.0000017014.52972.1d.
10. Tverdokhleb, E.; Dobrovolskyi, H.; Keberle, N.; Myronova, N. Implementation of accent recognition methods subsystem for eLearning systems. In Proceedings of the 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Bucharest, Romania, 21–23 September 2017; Volume 2, pp. 1037–1041. https://doi.org/10.1109/IDAACS.2017.8095243.
11. Ensslin, A.; Goorimoorthee, T.; Carleton, S.; Bulitko, V.; Poo Hernandez, S. Deep Learning for Speech Accent Detection in Video games. In Proceedings of the Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference, Salt Lake City, UT, USA, 5–9 October 2017; Volume 13.
12. Berjon, P.; Nag, A.; Dev, S. Analysis of French phonetic idiosyncrasies for accent recognition. *Soft Comput. Lett.* **2021**, *3*, 100018. https://doi.org/10.1016/J.SOCL.2021.100018.
13. Bird, J.; Wanner, E.; Ekárt, A.; Faria, D. Accent Classification in Human Speech Biometrics for Native and Non-native English Speakers. In Proceedings of the PErvasive Technologies Related to Assistive Environments (PETRA), Rhodes, Greece, 5–7 June 2019; pp. 554–560. https://doi.org/10.1145/3316782.3322780.
14. Zhang, Z.; Wang, Y.; Yang, J. Accent recognition with hybrid phonetic features. *Sensors* **2021**, *21*, 6258.
15. Malla, S.S. Acoustic Features Based Accent Classification of Kashmiri Language using Deep Learning. *Glob. J. Comput. Sci. Technol.* **2022**, *22, 39–43* .
16. Graham, C. L1 Identification from L2 Speech Using Neural Spectrogram Analysis. *Interspeech* **2021**, *2021*, 3959–3963. https://doi.org/10.21437/Interspeech.2021-1545.
17. Ahamad, A.; Anand, A.; Bhargava, P. AccentDB: A Database of Non-Native English Accents to Assist Neural Speech Recognition. *arXiv* **2020**, arXiv:2005.07973.
18. Oladipo, F.; Habeeb, R.A.; Musa, A.E. Accent Identification of Ethnically Diverse Nigerian English Speakers. *SSRN Electron. J.* **2020** . https://dx.doi.org/10.2139/ssrn.3666815.
19. Aswathi Sanal, M. Accent Recognition for Malayalam Speech Signals. *Int. J. Innov. Res. Comput. Commun. Eng.* **2017**, *5*, 4013–4017.
20. Ma, Y.; Paulraj, M.; Yaacob, S.; Shahriman, A.; Nataraj, S.K. Speaker accent recognition through statistical descriptors of Mel-bands spectral energy and neural network model. In Proceedings of the 2012 IEEE Conference on Sustainable Utilization and Development in Engineering and Technology (STUDENT), Kuala Lumpur, Malaysia, 6–9 October 2012; pp. 262–267. https://doi.org/10.1109/STUDENT.2012.6408416.
21. Duong, Q.T.; Do, V.H. Development of Accent Recognition Systems for Vietnamese Speech. In Proceedings of the 2021 24th Conference of the Oriental COCOSDA International Committee for the Co-ordination and Standardisation of Speech Databases and Assessment Techniques (O-COCOSDA), Singapore, 18–20 November 2021; pp. 174–179.
22. Krishna, G.R.; Krishnan, R.; Mittal, V.K. A system for automatic regional accent classification. In Proceedings of the 2020 IEEE 17th India Council International Conference (INDICON), New Delhi, India,10–13 December 2020; pp. 1–5.
23. Cheng, J.; Bojja, N.; Chen, X. Automatic accent quantification of Indian speakers of English. In Proceedings of the Interspeech, Lyon, France, 25–29 August 2013; pp. 2574–2578.
24. Lazaridis, A.; el Khoury, E.; Goldman, J.P.; Avanzi, M.; Marcel, S.; Garner, P.N. Swiss French Regional Accent Identification. In Proceedings of the Odyssey, Joensuu, Finland, 16–19 June 2014; pp. 106–111.

25. Jiao, Y.; Tu, M.; Berisha, V.; Liss, J.M. Accent Identification by Combining Deep Neural Networks and Recurrent Neural Networks Trained on Long and Short Term Features. In Proceedings of the Interspeech, San Francisco, CA, USA, 8–12 September 2016; pp. 2388–2392.
26. Weninger, F.; Sun, Y.; Park, J.; Willett, D.; Zhan, P. Deep Learning Based Mandarin Accent Identification for Accent Robust ASR. In Proceedings of the Interspeech, Graz, Austria, 15–19 September 2019; pp. 510–514.
27. Işik, G.; Artuner, H. Turkish Dialect Recognition Using Acoustic and Phonotactic Features in Deep Learning Architectures. *J. Inf. Technol.* **2020**, *13*, 207–216.
28. Kethireddy, R.; Kadiri, S.R.; Alku, P.; Gangashetty, S.V. Mel-Weighted Single Frequency Filtering Spectrogram for Dialect Identification. *IEEE Access* **2020**, *8*, 174871–174879.
29. George Mason University. Speech Accent Archive, 2021. Available online: https://accent.gmu.edu/ (accessed on 8 August 2022).
30. Zheng, F.; Zhang, G.; Song, Z. Comparison of different implementations of MFCC. *J. Comput. Sci. Technol.* **2001**, *16*, 582–589.
31. Liu, S.; Wang, D.; Cao, Y.; Sun, L.; Wu, X.; Kang, S.; Wu, Z.; Liu, X.; Su, D.; Yu, D.; et al. End-to-end accent conversion without using native utterances. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6289–6293.
32. Rasier, L.; Hiligsmann, P. Prosodic transfer from L1 to L2. Theoretical and methodological issues. *Nouv. Cah. Linguist. Française* **2007**, *28*, 41–66.
33. Alain de Cheveigné, H.K. YIN, a fundamental frequency estimator for speech and music. *J. Acoust. Soc. Am.* **2002**, *111*, 1917–1930. https://doi.org/10.1121/1.1458024.
34. Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *J. Stat. Plan. Inference* **2000**, *90*, 227–244. https://doi.org/https://doi.org/10.1016/S0378-3758(00)00115-4.
35. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; Bach, F., Blei, D., Eds.; PMLR: Lille, France, 2015; Volume 37, pp. 448–456.
36. Moreno-Torres, J.G.; Raeder, T.; Alaiz-Rodríguez, R.; Chawla, N.V.; Herrera, F. A unifying view on dataset shift in classification. *Pattern Recognit.* **2012**, *45*, 521–530. https://doi.org/https://doi.org/10.1016/j.patcog.2011.06.019.
37. Y, G.D.; Nair, N.G.; Satpathy, P.; Christopher, J. Covariate Shift: A Review and Analysis on Classifiers. In Proceedings of the 2019 Global Conference for Advancement in Technology (GCAT), Bangalore, India, 18–20 October 2019; pp. 1–6. https://doi.org/10.1109/GCAT47503.2019.8978471.
38. Bock, S.; Weiß, M. A proof of local convergence for the Adam optimizer. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
39. Longford, N.T. A fast scoring algorithm for maximum likelihood estimation in unbalanced mixed models with nested random effects. *Biometrika* **1987**, *74*, 817–827.
40. Wu, T.; Duchateau, J.; Martens, J.P.; Van Compernolle, D. Feature subset selection for improved native accent identification. *Speech Commun.* **2010**, *52*, 83–98.
41. Sun, L.; Wang, T.; Ding, W.; Xu, J.; Lin, Y. Feature selection using Fisher score and multilabel neighborhood rough sets for multilabel classification. *Inf. Sci.* **2021**, *578*, 887–912.
42. Milner, B.; Shao, X. Prediction of Fundamental Frequency and Voicing From Mel-Frequency Cepstral Coefficients for Unconstrained Speech Reconstruction. *IEEE Trans. Audio Speech Lang. Process.* **2007**, *15*, 24–33. https://doi.org/10.1109/TASL.2006.876880.
43. Bogach, N.; Boitsova, E.; Chernonog, S.; Lamtev, A.; Lesnichaya, M.; Lezhenin, I.; Novopashenny, A.; Svechnikov, R.; Tsikach, D.; Vasiliev, K.; et al. Speech Processing for Language Learning: A Practical Approach to Computer-Assisted Pronunciation Teaching. *Electronics* **2021**, *10*, 235.
44. Mikhailava, V.; Blake, J.; Pyshkin, E.; Bogach, N.; Chernonog, S.; Zhuikov, A.; Lesnichaya, M.; Lezhenin, I.; Svechnikov, R. Dynamic Assessment during Suprasegmental Training with Mobile CAPT. *Proc. Speech Prosody* **2022**, *2022*, 430–434.
45. Feng, Y.; Fu, G.; Chen, Q.; Chen, K. SED-MDD: Towards sentence dependent end-to-end mispronunciation detection and diagnosis. In Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 3492–3496.