Automatic Language Identification from Audio Signals using LSTM-RNN

Batir Sharimbaev
Mathematics and Natural Sciences
Suleyman Demirel University
Kaskelen, Kazakhstan
batyr.sharimbaev@gmail.com

Shirali Kadyrov
Mathematics and Natural Sciences
Suleyman Demirel University
Kaskelen, Kazakhstan
shirali.kadyrov@sdu.edu.kz

2023 17th International Conference on Electronics Computer and Computation (ICECCO) | 979-8-3503-3961-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICECCO58239.2023.10146603

***Abstract*— The objective of this study is to develop an efficient Language Identification (LID) system using Long Short-Term Memory Recurrent Neural Networks applied to audio signals. Two experiments were conducted to validate the proposed approach. The experimental results demonstrated exceptional performance, with an accuracy of 98% and 97.6% on the test sets of the first and second experiments, respectively. The models were trained and tested using audio recordings in English, Russian, Turkish, Kyrgyz, and Kazakh languages. These findings suggest that the proposed LID system is highly effective and can be used in various real-world applications.**

Keywords—RNN, LSTM, Language Identification, Audio Signals.

I. INTRODUCTION

Automatic speech recognition systems are extensively utilized in human-computer interaction, encompassing various applications such as voice assistants like Alexa, Alisa and Google Assistant, voice recognition, smart systems, and transcribing recorded audio. Recognizing and detecting spoken language is the primary task to make speech recognition systems more inclusive. Real-time control of machines and inputting information through human speech greatly simplifies modern life. Numerous models have been trained and tested to support this objective.

Several popular techniques have been developed for automated language identification (LID), such as N-gram models [9, 10], Naive Bayes classifiers [11], Support Vector Machines (SVMs) [12], and Deep Neural Networks (DNNs) [1, 3]. Among these, Convolutional Recurrent Neural Networks (CRNN) and Long Short-Term Memory (LSTM) Recurrent Neural Networks are widely used in DNNs and have proven to be effective in achieving high accuracy in LID tasks. For example, in the work [1], collected a dataset from the EU Speech Repository and YouTube news. They converted all audio files to WAV format, extracted the audio data into spectrogram representations, and saved them as lossless 500×129 PNG files. The network in this study was implemented using Keras deep learning framework. The datasets were divided into a training set (70%), a validation set (20%), and a testing set (10%) to evaluate the performance of the network. The Adam optimizer was used to train the networks, and Stochastic Gradient Descent was utilized to fine-tune them. the metric accuracy, recall, precision, and F1 score were reported. The highest score was 96% for Inception-v3 CRNN. In another work [2], model was trained and tested using three languages: Yoruba, Igbo and English. Dataset consists of over 2100 audio clips, 700 samples for each language. 80% of the sample files are used for train set and 20% of data are used for validation and test sets. The model's accuracy for more than 100 Epochs was given as 95.56%.

In a recent work [3], the six languages of English, German, French, Spanish, Italian, and Greek were used in various sets, with comparable Convolutional Recurrent Neural Network architecture. Collected audio clips from The European Speech Repository, Youtube News and VoxForge. The architecture of Model consists CNN and CRNN model. "ConvBlock" consists of a 2D convolutional layer, BatchNormalization layer and a Max Pooling layer with a pool size and step size of 2x2. For N classes, the output layer has a softmax activation function. The higher accuracy for CRNN architecture is 98% classification for training set and 83% of classification for testing set. Also, they trained CNN model on Youtube News dataset and tested on VoxForge. The classification accuracy of this experiment was 32%.

Recurrent Neural Network (RNN) with Long Short-Term Memory Networks (LSTMs) are popular choice for processing audio signals in fields such as speech recognition, speech synthesis, and music generation. In a study [4] authors used the NIST Language Recognition Evaluation 2009 (LRE'09) dataset. To ensure that the experiments were conducted accurately, the three different sets of languages were differentiated. The target languages include US English, Spanish, Dari, French, Pashto, Russian, Urdu, and Chinese Mandarin. The accuracy of the system was evaluated using audio clips ranging from 0.1 to 2.25 seconds, which were determined by Voice Activity Detection (VAD). The accuracy over 70% for 2 seconds and 0.5 seconds are needed for 50% accuracy. The same dataset also used in the work [5] with 20 times fewer parameters. LSTM RNN architecture outperforms the top four layer DNN system in terms of performance. Additionally, discovered that LSTM RNN scores were more accurately calibrated than those generated by the i-vector or DNN systems. In terms of Equal Error Rate (EER) and the average classification score (C_avg) , DNNs and LSTM RNN both outperform i-vector systems by 47% and 52%, respectively. The results of the i-vector system resulted in a 28% improvement overall.

The work [6] used three code-switched language pairs Gujarati-English, Tamil-English and Telugu-English. Model has five LSTM layers with a total of 1024 neurons and used the Connectionist Temporal Classification (CTC) loss function. The LID utilizing CTC begins with two layers of 2D convolutions in the time and frequency domains with filter dimensions of 32, 41×11, 21×11 with a stride of 2×2, 2×1. Five BLSTM layers with softmax outputs will follow. The accuracy of the model on different datasets 71-73% in Gujarati-English, 71-79% of classification in Tamil-English and 74-78% accuracy in Telugu-English.

The last but not the least, Kaiyr et al. [7] proposed a novel method for automatic language identification from spectrogram images using deep neural networks. They applied

convolutional neural network (CNN) and long short-term memory (LSTM) algorithms on the preprocessed audio data converted to spectrograms. They experimented with seven languages: English, Kazakh, French, German, Italian, Russian and Spanish. They achieved over 99% training accuracy and 94,28% testing accuracy on their dataset. Their findings indicated that their model was able to differentiate Kazakh language from others with 100% accuracy. However, their method was not tested on languages that are close to Kazakh, which they found to be easily distinguishable from the others due to its different language family and dataset source. Therefore, in the first experiment, we focus on the three most widely spoken languages in Kazakhstan: English, Russian and Kazakh. For the second experiment, we aim to extend their method by adding two more languages that are similar to Kazakh: Kyrgyz and Turkish. We will investigate how LSTM based algorithms performs on these languages and whether it can still maintain its high accuracy.

METHODOLOGY

A. Data Collection

We are interested in the task of identifying five different languages: English, Turkish, Russian, Kyrgyz, and Kazakh. However, since there are no readily available, large-scale datasets for automated Language Identification (LID) tasks, we have created our own dataset for experimentation.

English audio files were downloaded from The Microsoft Scalable Noisy Speech Dataset (MS SNSD) [13]. Russian and Turkish audio files were collected from VoxForge [14]. Kyrgyz recordings were taken from Mozilla Common Voice [15]. Kazakh audio files were collected by converting YouTube videos to audio files. The speakers in the dataset had varying accents and genders, and some speakers may have appeared in multiple audio clips. However, the training, testing, and validation sets were kept free from any cross-contamination. The audio files were saved in WAV format and imported into Python using librosa [8] with a sample rate of 22,050 Hz. All 2500 audio recordings compose of the entire dataset, 500 samples in all English, Turkish, Russian, Kyrgyz and Kazakh languages. The YouTube data was converted into WAV files and then segmented into short, noise-free clips, with each audio sample lasting between 5 and 15 seconds. Noises have not been cleaned from them. Importantly, only the first 3 seconds of audio clips were used for collected Mel-frequency cepstral coefficients (MFCCs).

The obtained audio data has many desired properties. The dataset contains languages that are very close to each other. They are Kazakh, Kyrgyz and Turkish languages. This is to know how well the Language Identification (LID) program detects the language. Importantly, audio recordings were provided as MFCCs.

B. Data Preprocessing

We performed some preprocessing on the collected data to ensure compatibility. Firstly, English, Turkish, Russian, Kyrgyz were downloaded from open source datasets and converted to wav format. Kazakh audio files taken from News, a lot of different lessons, history channels of YouTube.

Next, we loaded the audio files using the os and librosa libraries, and ensured that their lengths were consistent with each other by cutting them. The audio files were recorded at a sampling rate of 22,500 Hz, and each file corresponds to three second of audio. However, if the total length of the audio clip is less than 67,500 Hz, we exclude it from our analysis.. Package os was used to get details from the audios in the file. These details are: mappings, labels, MFCCs and filenames of audios. Then, by using package librosa, extracted MFCCs. In this process are done 2048 samples in the fast Fourier transform. Number of MFFCs is 13 and 512 samples between overlapping windows. Audio signals' mappings, labels, MFCCs and filenames sre stored into dictionaries and saved as JSON file.

C. Neural Networks

Recurrent Neural Networks (RNNs) are commonly used for audio analysis models. RNNs used for sequential data and each item is processed in context. The Neural Networks are considered ideal for audio or music. One of the best common architectures in Recurrent Neural Network is Long Short-Term Memory (LSTM) Network which processes entire sequence of data. LSTM Network learn long-term patterns and detects its with 100 steps.

D. Model of Experiment 1

We propose Long Short-Term Memory Recurrent Neural Network model for language identification. The model summary presented in Table 1 was implemented using Keras. There are 57,347 trainable parameters and not non-trainable parameters. During the build network topology we used 2 LSTM layers with Tanh activation tanh for LSTM Network and Recurrent Neural Networks used sigmoid activation sigmoid function. Units of the layers are 64, 64. After that, dense layer with sigmoid activation and 64 units. Dropout with probability 30% is applied to the dense layer to avoid over fitting. First LSTM and dense layers used L2 regularization with parameter 0,001 for solve overfitting. The final layer utilizes a softmax activation function with 3 units.. 80% of the data is given to the train set and 20% to the test and validation sets, it is divided from 10% to each set.

Table 1. Model Summary of Experiment 1

Layer	Output (shape)	Params #
LSTM	(None, 44, 64)	19 968
LSTM	(None, 64)	33 024
Dropout	(None, 64)	0
Dense	(None, 64)	4 160
Dropout	(None, 64)	0
Dense	(None, 3)	195

E. Model of Experiment 2

The model experiment was similar to Experiment 1, with only a few parameter changes. Model summary showed in Table 2. The model used 61,637 trainable parameters and not non-trainable parameters. We used 2 LSTM layers with default activation functions and 64 of units. More specifically, the model has 2 dense layers with sigmoid activation and 64 of units. In order to combat overfitting, the first LSTM layer and two dense layers undergo L2

regularization with a parameter of 0.001 and Dropout with probabilities of 30%. The output layer is configured with softmax activation and 5 units, which enables it to classify the five classes of audio.

Table 2. Model Summary of Experiment 2

Layer	Output (shape)	Params #
LSTM	(None, 44, 64)	19 968
LSTM	(None, 64)	33 024
Dropout	(None, 64)	0
Dense	(None, 64)	4 160
Dense	(None, 64)	4 160
Dropout	(None, 64)	0
Dense	(None, 5)	195

II. Results of the Experiments

A. Experiment 1

The purpose of this experiment is to create a Language Identification (LID) model using data from the three most popular languages in Kazakhstan: English, Russian, and Kazakh. A total of 1500 recordings were used, with 500 recordings for each language. For each recording, only the first three seconds of audio were analyzed.

Figure 1 shows the accuracy graphs for the training and validation sets, which both achieved 100% accuracy. The test set had a 98% classification accuracy. The model performed well over 100 epochs of fitting, with steady improvement and no noticeable deviations. The graph indicates accuracies above 99% after 20 epochs, but we increased the number of epochs to further reduce errors.

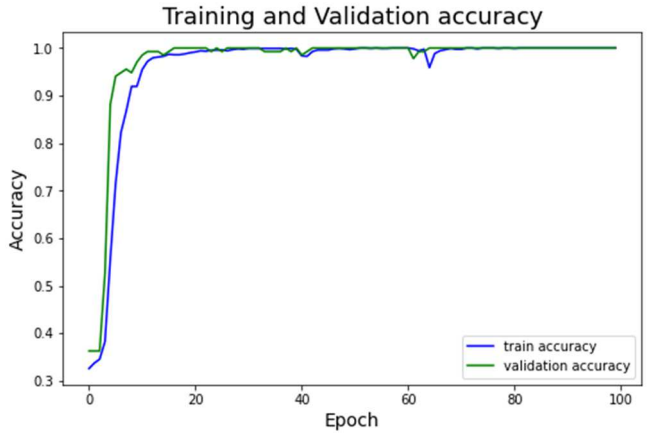

Figure 1. The graph of accuracy of Experiment 1

Figure 2 shows the progress of Categorical Cross Entropy (CCE) cost function values for both the train and validation sets. Errors decreased very quickly and stable. The cost function used for the train and test sets gave a value of approximately 0.05. The progress of errors have not any fluctuations. The cost function of the test set was 0.1.

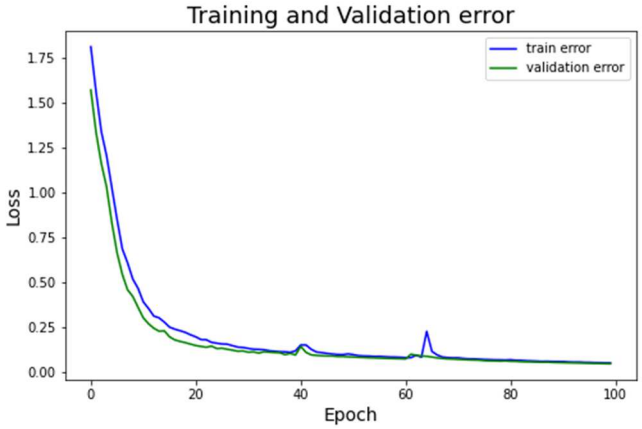

Figure 2. The graph of errors of Experiment 1

The confusion matrix for the classification was displayed in Figure 3. Here we can see prediction of test set for each language. The labels on the left side of the matrix are correct values and predictions are located on the bottom. The values of the diagonals are the number of values correctly found by the model. Confusion matrix shows very good results. Predictions for Kazakh language gave 2 mistakes and to find audio clips in Russian was 1 error. All English audio recordings are correctly predicted.

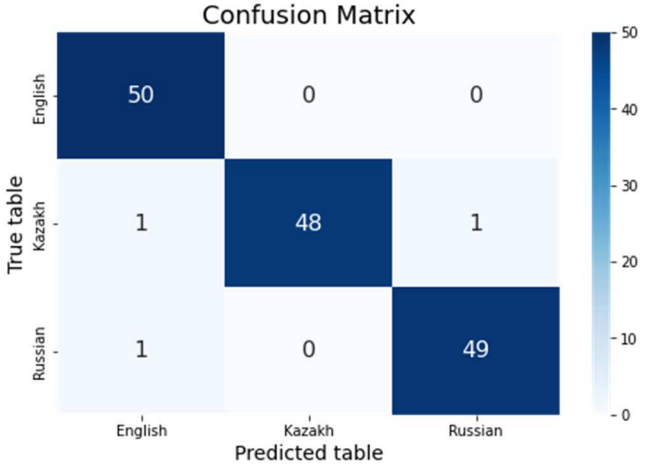

Figure 3. Confusion Matrix of Experiment 1

Figure 4 displays the model's prediction accuracy for each language, obtained from the confusion matrix. The histogram shows that English was classified with 100% accuracy, while Russian had a 98% accuracy rate. The model achieved 96% accuracy in identifying Kazakh audio recordings.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

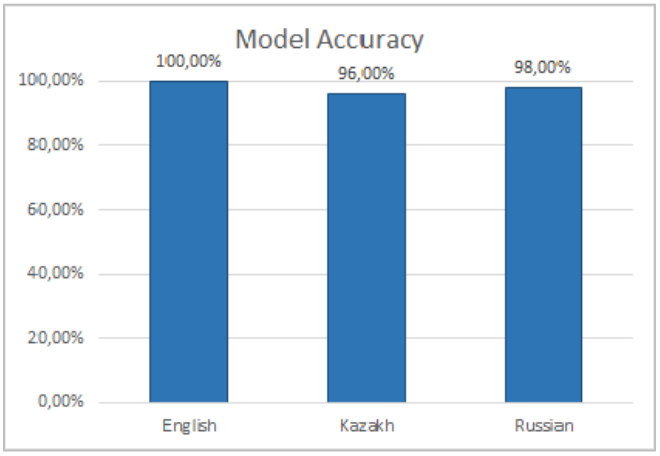

Figure 4. Histogram of model accuracy of Experiment 1

Additionally, we considered at which audio clips were incorrectly predicted. 3 wrong predictions were made in the experiment. We found the audio files and tried listening to audios using the iPython package. While analyzing the audios, we noticed a gap in the first 3 seconds in one of the audio recordings. We believe that the model made a wrong decision since it was unable to extract any information from the audio.

B. Experiment 2

Building upon the success of Experiment 1, we expanded our dataset to include two additional languages. To test the model's ability to identify closely related languages, we selected Kyrgyz and Turkish, which share similarities with the Kazakh language. Our dataset now consists of 2500 audio recordings, with 500 recordings for each language.

The graph of training and validation accuracy is shown in Figure 5. Maximum results of train set showed higher than 96% accuracy and 94% of classification on the validation set. The identification accuracy of test set was 97.6%. The progress of train and validation accuracy were not any fluctuations.

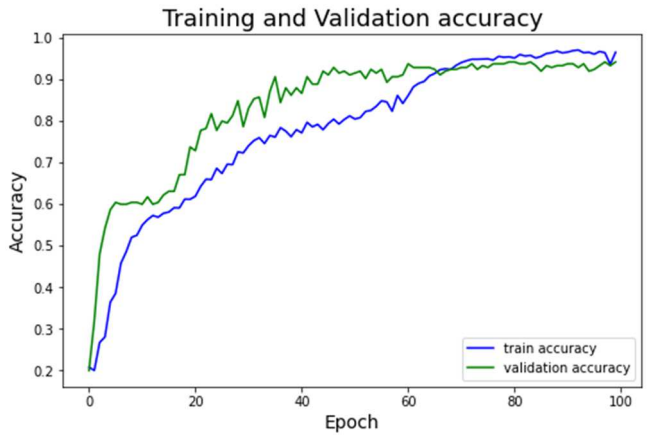

Figure 5. The graph of accuracy of Experiment 2

Figure 6 shows the cost function graphs for both the training and test sets. The model performs exceptionally well, with errors decreasing below 0.31. The values of errors are stable decrease. It can be said that there were no deviations at all. Result of cost function for test set was 0.26.

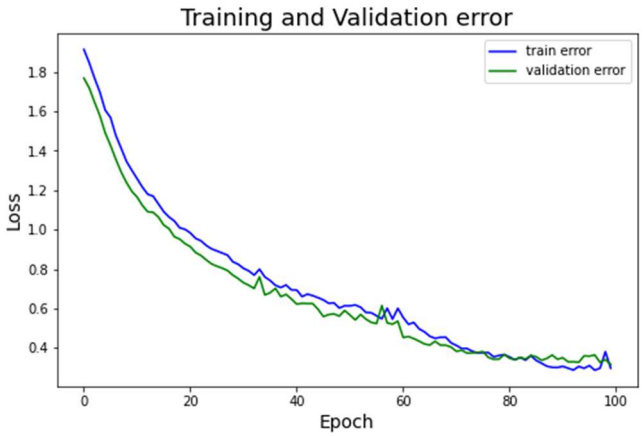

Figure 6. The graph of errors of Experiment 2

The confusion matrix of experiment 2 is shown in Figure 7. There are no noticeable errors in the matrix. Let's recall that we took five languages to implement this experiment. They are English, Russian, Turkish, Kyrgyz and Kazakh languages. All English and Russian audio clips in the test set are predicted correctly. Because, the languages are completely different from the languages we use for the experiment. As for the errors made during prediction, there were 1 error in Kazakh, 3 in Kyrgyz and 2 errors in Turkish. If we look closely at the matrix, the mistakes were only in the Kyrgyz language. Based on information on the Internet, the language is closer to the Turkish language than the Kazakh language. In addition, Kazakh and Kyrgyz words are very similar. Therefore, we think that the model made small errors in language detection.

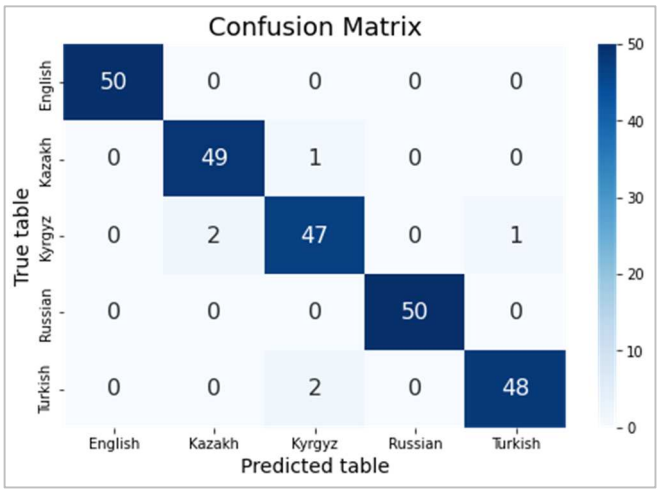

Figure 7. Confusion Matrix of Experiment 2

Using the accuracy formula and the confusion matrix given in Figure 7, we created the histogram in Figure 8. In machine learning or deep learning, an accuracy above 90% is considered the best accuracy score. Here, all languages are above 90% accuracy and we can see 100% identification for English, Russian languages. Based on this information, it can be said that this experiment was successfully performed.

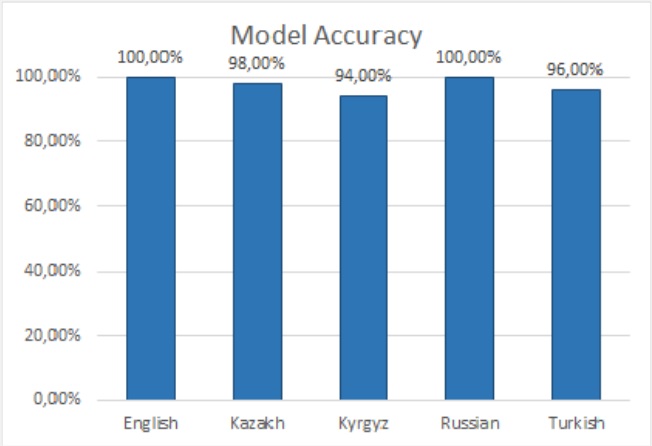

Figure 8. Histogram of Model accuracy of Experiment 2

In the end, we found the errors in this experiment. Noises and interruptions were observed in some audio clips. Perhaps, without these obstacles, the model would have given better results.

III. CONCLUSION

In this work, we implemented Language Identifier from audio clipls using Long Short-Term Memory Recurrent Neural Networks. For the dataset, we clipped the first 3 seconds of audio and converted to MFCCs. Five languages English, Russian, Turkish, Kyrgyz and Kazakh were considered in the classification. We did 2 experiments and the experiments showed best results. In experiment 1, the training set showed 100% accuaracy, and 98% of identification on the test set. Experiment 2 showed 96% of classification on the training set and 97,6% accuracy on the testing set. We can say that the results are very good, because during operation the noise is not removed from the sounds and the audios last only 3 seconds. 10% of the data is allocated for test set, 80% is used to train the network and 10% of data for validation set. 500 dates were obtained for each language.

The purpose of the first experiment was to propose language identifier for popular languages (English, Russian, Kazakh) in Kazakhstan, while the second experiment was to investigate how well language identifier identify similar languages (Kazakh, Kyrgyz, Turkish). If look at Figure 8, we can see that English and Russian are 100% classified. Because those languages are not similar to the languages we use.

In future work, we will use a large dataset to create model that solves the real-world problem.

REFERENCES

[1] Christian Bartz et al. “Language identification using deep convolutional recurrent neural networks”. In: Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14–18, 2017, Proceedings, Part VI 24. Springer. 2017, pp. 880–889.

[2] Nancy Woods and Gideon Babatunde. “A robust ensemble model for spoken language recognition”. In: Applied Computer Science 16.3 (2020), pp. 56–68.

[3] Alexandra Draghici, Jakob Abeßer, and Hanna Lukashevich. “A study on spoken language identification using deep neural networks”. In: Proceedings of the 15th International Audio Mostly Conference. 2020, pp. 253–256.

[4] Ruben Zazo et al. “Language identification in short utterances using long short-term memory (LSTM) recurrent neural networks”. In: PloS one 11.1 (2016), e0146917.

[5] Javier Gonzalez-Dominguez, Ignacio Lopez-Moreno, and Hasim Sak. “Automatic language identification using long short-term memory recurrent neural networks”. In: (2014).

[6] Pradeep Rangan, Sundeep Teki, and Hemant Misra. “Exploiting spectral augmentation for code-switched spoken language identification”. In: arXiv preprint arXiv:2010.07130 (2020).

[7] Kaiyr, A., Kadyrov, S. and Bogdanchikov, A., 2021, April. Automatic Language Identification from Spectorgam Images. In 2021 IEEE International Conference on Smart Information Systems and Technologies (SIST) (pp. 1-4). IEEE.

[8] Brian McFee et al. “librosa: Audio and music signal analysis in python”. In: Proceedings of the 14th python in science conference. Vol. 8. 2015, pp. 18–25.

[9] Tommi Vatanen, Jaakko J Väyrynen, and Sami Virpioja. “Language Identification of Short Text Segments with N-gram Models.” In: LREC. 2010.

[10] Katrin Kirchhoff and Sonia Parandekar. “Multi-stream statistical n-gram modeling with application to automatic language identification”. In: Seventh European Conference on Speech Communication and Technology. 2001.

[11] Pablo Gamallo et al. “Comparing Ranking-based and Naive Bayes Approaches to Language Detection on Tweets.” In: TweetLID@ SEPLN. 2014, pp. 12–16.

[12] William M Campbell et al. “Language recognition with support vector machines”. In: ODYSSEY04-The Speaker and Language Recognition Workshop. 2004.

[13] Chandan KA Reddy et al. “A scalable noisy speech dataset and online subjective test framework”. In: arXiv preprint arXiv:1909.08050 (2019).

[14] Khaled Lounnas et al. “A language identification system based on voxforge speech corpus”. In: The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2019). Springer. 2019, pp. 529–534

[15] Samiul Alam et al. “Bengali common voice speech dataset for automatic speech recognition”. In: arXiv preprint arXiv:2206.14053 (2022).