MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE REPUBLIC OF KAZAKHSTAN

INTERNATIONAL INFORMATION TECHNOLOGY UNIVERSITY

FACULTY OF COMPUTER TECHNOLOGY AND CYBERSECURITY

**Menglibayev Z.E.**

**Askarbekov A.S.**

**Khakhazov T.Kh.**

**Speaker verification using machine and deep learning methods**

**DIPLOMA PROJECT**

**Majors 6B06108** – **Data Science and Machine Learning,**

**6B06110** – **Software engineering**

Almaty 2023

MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE REPUBLIC OF KAZAKHSTAN

INTERNATIONAL INFORMATION TECHNOLOGY UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

**Approved**

Head of Department,

Assistant prof., PhD,

Chinibayeva T.T.

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023

**DIPLOMA PAPER**

**Speaker verification using machine and deep learning methods**

Majors 6B06108 – Data Science and Machine Learning,

6B06110 – Software engineering

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Done by: |  | Menglibayev Z.E. |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
|  |  | «\_\_\_ » \_\_\_\_\_\_2023 |  | (signature) |  |
|  |  | Askarbekov A.S. |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
|  |  | «\_\_\_ » \_\_\_\_\_\_2023 |  | (signature) |  |
|  |  | Khakhazov T.Kh. |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
|  |  | «\_\_\_ » \_\_\_\_\_\_2023 |  | (signature) |  |
|  |  |  |  |  |  |
| Research advisor: |  | Sarsembayev A.A. |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
|  |  | «\_\_\_ » \_\_\_\_\_\_2023 |  | (signature) |  |
| Reviewer: |  | Abeshev K. Sh. |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
|  |  | «\_\_\_ » \_\_\_\_\_\_2023 |  | (signature) |  |

Almaty 2023

International Information Technology University

Faculty of Computer Technology and Cyber Security

Department of Computer Engineering

Majors 6B06108 – Data Science and Machine Learning,

6B06110 – Software engineering

Diploma Project Assignment

Students

**Menglibayev Z.E. Askarbekov A.S. Khakhazov T.Kh.**

Diploma project topic

**Speaker verification using machine and deep learning methods**

Approved by IITU order № \_\_\_ dated «\_\_\_» \_\_\_\_\_\_ 20\_\_

Diploma work (project) submission date «\_\_\_»\_\_\_\_ 20\_\_

Diploma work (project) initial data

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Details of computations and explanations (list of issues due to be addressed)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

CD containing the digital version of diploma paper and attachments

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Consultations on diploma work (project) (with related project chapters named)

|  |  |  |  |
| --- | --- | --- | --- |
| Consultant | Name | Signature, date | |
| Assignment given | Assignment received |
| Consultant on Economic  effectiveness of the project | Mukhamediyeva A.G., Candidate of Economic Sciences, associate professor |  |  |
| English language consultant | Bekbulatov T.R.,  Senior lecturer |  |  |
| [Compliance monitor](http://www.multitran.ru/c/m.exe?t=4330399_1_2&s1=%ED%EE%F0%EC%EE%EA%EE%ED%F2%F0%EE%EB%FC) | Mamanova S.E.,  Senior lecturer |  |  |

|  |  |
| --- | --- |
| Date «\_\_» \_\_\_\_\_\_20\_\_ |  |
| Research advisor | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (signature) |
| Assignment received by | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

(signature)

Diploma project writing schedule

**Menglibayev Z.E. Askarbekov A.S. Khakhazov T.Kh.**

Title: **Speaker verification using machine and deep learning methods**

|  |  |  |
| --- | --- | --- |
| № | Assignment | Submission date |
| 1. | Creation of the graduation paper writing schedule | November 30 |
| 2. | Collection, study, processing, analyzing, and generalizing data | November – December |
| 3. | Drafting and submission to the Research advisor  (Introduction, Chapter 1, Chapter 2, Chapter 3, Chapter 4, Conclusion) | January – February |
| 4. | Submission of the chapter «Economic effectiveness of the project» to the consultant | February – March |
| 5. | Revision of the graduation paper with due consideration of the advisor’s comments | March – April |
| 6. | Submission of the completed diploma paper to the Research advisor | April 15 |
| 7. | Pre-defence | April |
| 8. | Submission of the completed diploma paper to the English language consultant | April 20 – April 30 |
| 9. | Submission of the diploma paper to the compliance monitor | April 30 – May 15 |
| 10. | Submission of the diploma paper for the plagiarism check | May 3 – May 24 |
| 11. | Submission to the reviewer for approval | May 3 – May 24 |
| 12. | Diploma work(project) defense | May 24 - June 19 |

Student: Menglibayev Z.E. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(signature)

Student: Askarbekov A.S. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(signature)

Student: Khakhazov T.Kh. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(signature)

Research advisor: Sarsembayev A.A. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(signature)

Date «\_\_» \_\_\_\_\_20\_\_

АҢДАТПА

Қазіргі уақытта сөйлеушіні автоматты түрде тану, тұлғаны тексеру және патологиялық сөйлеуді болжау әдістері кеңінен қолданылуда. Сонымен қатар түрлі салаларда аудио деректерді өңдеудің маңыздылығы артты. Осыған орай, дипломдық жобаның мақсаты - машиналық және терең оқыту әдістерін пайдалана отырып, аудио деректерге негізделе тұлғаны тексеретін бағдарлама жазу. Тапсырманың мәнісі әртүрлі деректер мен архитектуралық параметрлер оған қоса терең оқыту әдісін қолдана отырып, аудио модельді тұлғаны анықтауда ең жоғарғы дәлдікке жеткізу. Бұл мақсатқа жету үшін біз өз бағдарламамызды Telegram бот жүйесінде тексеруге болатындай бағытты таңдадық. Біздің бағдарламамыздың басты тұтынушыларының бірі ол қызметкерлерді немесе клиенттерді растау барысында осы технологияны қажет ететін компаниялар мен ұйымдар.

Дипломдық жоба 61 беттен, 5 кестеден, 26 иллюстрациядан, 8 сілтемеден әрі 16 әдеби тізімнен тұрады.

Түйін сөздер: ТҰЛҒАНЫ ТЕКСЕРУ, ТЕРЕҢ ОҚЫТУ, КОНВОЛЮЦИОНДЫ НЕЙРОНДЫҚ ЖЕЛІ, СПИКЕРДІ ТАНУ.

АННОТАЦИЯ

В настоящее время важность обработки речи возросла в различных областях, таких как автоматическое распознавание речи, верификация личности и прогнозирование патологической речи. По этой причине целью дипломного проекта является написание программы для верификации личности на основе аудио данных с использованием методов машинного и глубокого обучения. Задачей является обучение аудио модели на различных данных и разными параметрами архитектуры включающее глубокое обучение для достижения наибольшей точности определения личности. Для достижения поставленной цели и задач был выполнен анализ данных и источников для определения наиболее эффективных методов. В работе мы будем тестировать нашу программу в мессенджере Telegram в виде бота способного натренироваться на полученных данных и проводить верификацию личности. Нашей целевой аудиторией будут компании и организации, которым потребуется данная технология для подтверждения сотрудников или клиентов в зависимости от назначения.

Дипломный проект содержит в себе 61 страниц, 5 таблиц, 26 иллюстраций, 8 ссылок и 16 источников использованной литературы.

Ключевые слова: ВЕРИФИКАЦИЯ ЛИЧНОСТИ, ГЛУБОКОЕ ОБУЧЕНИЕ, СВЕРТОЧНАЯ НЕЙРОННАЯ СЕТЬ, РАСПОЗНАВАНИЕ ПО ГОЛОСУ.

ABSTRACT

In today's world, speech processing is more significant than ever in areas like pathological speech prediction, speaker verification, and automatic speech recognition. For this reason, the goal of the diploma project is to write a program for speaker verification using machine and deep learning methods based on audio data. The task is to train an audio model on various data and different architecture parameters, including deep learning to achieve the greatest accuracy in determining the speaker. We examined data and sources to identify the most efficient techniques for achieving the aim and goals. In the project, we will test our program using a bot that can learn from the data received and carry out identity verification in the Telegram messenger. Companies and organizations that require this technology to check clients or staff, depending on the situation, will be our target market.

The diploma project contains 61 pages, 5 tables, 26 illustrations, 8 links and 16 references.

Keywords: SPEAKER VERIFICATION, DEEP LEARNING, CONVOLUTIONAL NEURAL NETWORK, SPEAKER RECOGNITION.

CONTENTS

|  |  |  |
| --- | --- | --- |
|  | List of terms and abbreviations | 10 |
|  | INTRODUCTION | 11 |
| 1 | THEORETICAL BACKGROUND OF THE PROBLEM | 12 |
| 1.1 | Subject area | 12 |
| 1.1.1 | History of subject area | 12 |
| 1.1.2 | Topic relevance | 13 |
| 1.2 | Types of biometrics | 14 |
| 1.3 | Speaker recognition | 15 |
| 1.3.1 | Types of text dependencies in Speaker Verification | 16 |
| 1.4 | Literature review | 16 |
| 1.5 | Neural Networks | 17 |
| 1.6 | Team members roles | 20 |
| 1.7 | Development Environment | 21 |
| 1.7.1 | Jupyter Notebook | 21 |
| 1.7.2 | PyCharm | 22 |
| 1.8 | Programming language | 23 |
| 1.9 | Teamwork tools | 26 |
| 1.9.1 | Google Colaboratory | 27 |
| 1.10 | Summary | 29 |
| 2 | ARCHITECTURE AND DESIGN | 30 |
| 2.1 | Risk mitigation | 30 |
| 2.2 | Software Lifecycle model | 30 |
| 2.3 | Project requirements | 33 |
| 2.3.1 | Documentation requirements | 33 |
| 2.3.2 | Functional requirements | 34 |
| 2.3.3 | Non-functional requirements | 34 |
| 2.3.4 | Design requirements | 34 |
| 2.4 | Diagrams | 35 |
| 2.4.1 | Class Diagram | 35 |
| 2.4.2 | Activity Diagram | 36 |
| 2.4.3 | Sequence Diagram | 37 |
| 2.4.4 | Component Diagram | 39 |
| 2.4.5 | Deployment Diagram | 39 |

|  |  |  |
| --- | --- | --- |
| 3 | IMPLEMENTATION | 41 |
| 3.1 | Use scenarios | 41 |
| 3.2 | Dataset | 45 |
| 3.3 | Key parts of model training and testing | 47 |
| 3.4 | The result | 49 |
| 4 | ECONOMIC EFFECTIVENESS OF THE PROJECT | 50 |
| 4.1 | Technical description | 50 |
| 4.2 | Market analysis | 50 |
| 4.3 | Marketing action plan | 52 |
| 4.4 | Calculation of economic efficiency | 54 |
|  | CONCLUSION | 57 |
|  | REFERENCES | 58 |
|  | APPENDIX | 60 |

LIST OF TERMS AND ABBREVIATIONS

SV – Speaker Verification

TD-SV - Text-Dependent Speaker Verification

TI-SV - Text-Independent Speaker Verification

CNN - Convolutional Neural Network

DNN - Deep Neural Network

G2E2 - Generalized end-to-end

T2E2 - Tuple-based end-to-end

PLDA - probabilistic linear discriminant analysis

NN - Neural Network

NLTK - Natural Language Toolkit

UML - Unified Modeling Language

ML - Machine Learning

DL - Deep Learning

LPC - Linear Predictive Coding

ANN - Artificial Neural Network

ASV - Automatic Speaker Verification

ASI - Automatic Speaker Identification

IDE - Integrated Development Environment

NLP - Natural Language Processing

INTRODUCTION

The topic of the diploma project is "Speaker Verification Using Machine and Deep Learning Methods".

The main goal of the project is to write a program that will have acceptable and tolerable metrics with high verification accuracy on a test statement that will belong to a particular speaker.

The relevance of the project is the increasing need for trustworthy and secure means of recognition and authentication. In addition, numerous measures and signals have also been suggested and researched for use in biometric recognition systems. Fingerprint, facial, and voice measurements are some of the most popular. In the same way, speaker recognition technology has appeared in a number of commercial items over the last decade. Accordingly, speaker verification as the subsystem of speaker recognition is used in several well-liked on-site applications, including access control to automobiles, residences, warehouses, computer terminals, etc. Speaker verification is commonly used in forensics investigation and customization, as well as in distant applications including telecom networks, databases, websites, e-trade, banking transactions, and other secret transactions. A human's speech signal comprises linguistic and physiological data that can be used to identify the speaker. Human speech signals are a strong form of communication that can convey a wealth of data, including gender, emotional characteristics, accent, and more. Researchers can recognize speakers using voiceprint recognition thanks to these distinctive characteristics.

The object of this project is technology of speaker recognition in the modern world.

The subject of this project is speaker verification that relies on audio data of a person and verifies that person using methods of machine and deep learning.

Despite recent advances in the field of speaker recognition and due to a lack of sufficient research, this area still needs refinements to improve the speaker verification model. Thus, the theoretical significance of the project lies in its potential to accurately authenticate the identity of a speaker using machine learning and audio data.

The practical significance of the project is to build a model using various new ways of processing and training on audio data and use them in the verification of the speaker, given the prospects of this area in the future.

To achieve this goal, it is necessary to solve the following tasks:

1. Collect an acceptable matched dataset.

2. Train the model with different architectures.

3. Analyze the obtained metrics.

4. Get the best metric among others.

5. Determine the value of speaker recognition concepts.

6. Achieve a strong verification result.

7. Test our model on a specific platform.

1 THEORETICAL BACKGROUND OF THE PROBLEM

1.1 Subject area

The modern world does not stand still, so every day new technologies are introduced into our lives, and the old ones are improved. In addition, the significance of biometrics in our daily lives has grown considerably as this technology is being used in various fields. As an example, the fingerprints and facial identification of an individual in our world have significance in confirming some decisions and actions, and for other particular purposes. Hence, the initial key aspect is to comprehend the concept of biometric data and its potential applications.

1.1.1 History of subject area

Biometrics is the measurement and statistical analysis of people's unique physical and behavioral characteristics [1]. In line with this definition, biometric information can be used to identify a specific person. The history of the development of biometrics, according to some estimates, began already three millennia ago, when the inhabitants of ancient Babylon and Assyria knew that the leather pattern on the fingertips is different for everyone. In China, according to the research of Joe de Barros, biometrics was used already in the 14th century, and fingerprint identification is considered to be its most ancient manifestation. Chinese sellers used fingerprints to sign trade agreements to distinguish young children from each other. To do this, they dipped their fingers and palms in ink.

Biometrics finally developed in the 19th century. Francis Galton, Karl Pearson and Ronald Fisher are considered its theorists. Almost up to this time, personality recognition was based on fixing the image of a person in memory, but in the 1890s Alphonse Bertillon proposed a technique for measuring body parameters - Bertillonage, but the effectiveness of this method became ineffective when it became clear that different people can have the same parameters. The first person to draw attention to fingerprints as a form of identification in forensic science was Dr. Henry Faulds.[2]. Hence, nowadays we use all such data not only as physical things but also as digital data due to the advancement of computer technology.

Thanks to the development of high technologies, biometrics has become widely used in many fields of activity. Authentication, identification, ID card, payment for purchases, search for criminals and missing persons - this is just a small list of tasks that biometric methods are designed to solve now. The most common biometric methods today are fingerprint, iris, facial image, hand and palm vein pattern, and voice.

1.1.2 Topic relevance

In order to achieve the current results in the verification of the speaker, it is important to note the weight of the topic of speech recognition, the subsequent development of which has led to the greatest results.

To achieve satisfactory results in the field of speech recognition, work on this topic began in the middle of the last century. And the very first related system was Audrey, built by Bell Laboratories in the early 1950s. The main feature of the system was the identification of numbers spoken by only one person, but at that time this system could not work with speech.

In the 1960s first speaker recognition systems were published and used in the US military. However, the technology was limited by the lack of computational power and was limited in accuracy score.[13]

Further development in this area led to the 70s when new methods were identified and began to be applied. One of them was the method of dynamic programming and the method of linear prediction, which was used in Bell Laboratories systems. Moreover, in the 80s started the usage of hidden Markov models and artificial neural networks techniques, which significantly improved speech recognition systems' accuracy.

Additionally, research on speaker recognition continued to advance, creating new neural network models and investigating the use of Gaussian mixture models (GMMs), which are still widely employed today. But the dawn of technology came, of course, with the 2000s of the development of deep learning techniques, such as convolutional neural networks and recurrent neural networks giving high accuracy on recognition.[15]

Google developed Voice Search, a voice-activated iOS app, in 2008. With the advent of smartphones, voice control has a wide range of uses. You may use voice commands to use the many smartphone features with actual virtual assistants. Later, Google began using Android with its voice assistant. This assistant gives users the opportunity to access their own personalized content and preferences.

When Apple unveiled Siri, the first personal assistant with a recognizable voice, in 2011, it was using its own technology. As a result of developing this sphere, other tech behemoths bought speech assistants as well, proposing a different technology of verification.

In 2014, Amazon and Microsoft both launched Alexa with the option to register user voices. It may identify individual users' voices and provide customized content in response, such as music playlists, and audiobooks. [16]. As shown in Figure 1.1 experts predict that the global market for voice recognition technologies will reach $27.16 billion in value by 2025. Moreover, statistics show that in 2019 it was $10.7 billion, and in 2027 it is anticipated to be USD $28.1 billion.[3].



Figure 1.1 – Voice Recognition market growth

1.2 Types of biometrics

In Fig. 1.2, two different biometric techniques are depicted. Physiological biometrics are one type of biometric that can be utilized for identification or verification. Finding out a person's identity is referred to as identification. Investigations into illegal activity frequently use this technique. The alternative kind is behavioral biometrics. It serves as a means of verification. Verification is the procedure used to make sure a person is who they say they are. This approach examines trends in how people carry out particular tasks.

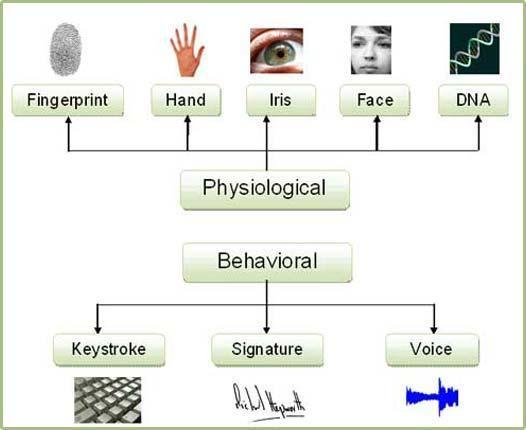


Figure 1.2 – Biometric Classification

The physical components of a person's voice, which are utilized to produce sound, include their vocal tracts, mouth, nasal cavities, and lips. These aspects of human speech are constant for a particular person, but the behavioral aspect varies over time owing to aging, illnesses, and emotional states. Two broad approaches are used to categorize voice recognition techniques: 1) Automatic Speaker Verification and 2) Automatic Speaker Identification.

1.3 Speaker recognition

Thanks to the development of personality voice processing, humanity has opened a new path of development in biometrics. With the development of sound recording and reproducing technology, speaker recognition technology is used with varying degrees of success in the field of information security, security and access systems, and forensics. And at this stage it will be correct to give the correct definition of the word “speaker recognition”. A Speaker Recognition (SR) system evaluates a person's voice or speech characteristics in order to determine who they are.

Speaker recognition can be divided into speaker identification and speaker verification. Speaker verification is defined as follows: “The goal is to determine if a test utterance belongs to a single particular speaker. It can be thought of as a template matching, thus the decision is binary, the system can accept or reject the utterance.” [5] Similarly, speaker identification is defined as follows: “It determines an anonymous speaker’s identity depending on the speaker’s spoken utterances. Speaker identification finds the exact speaker from a set of recognized voices of speakers. It is the way to find a person based on the different utterances contained in the database.” [6]

1.3.1 Text-Dependent and Text-Independent Speaker Verification

There are two types of speaker verification: text-dependent and text-independent. According to [7] text-independent is: “In text-independent systems, there are no constraints on the words which the speakers are allowed to use. Thus, the reference and the test utterances may have completely different content, and the recognition system must take this phonetic mismatch into account.” And according to [8] “Text-dependent speaker verification aims to compensate for phonetic variability, which poses a significant challenge in speaker verification”.

1.4 Literature review

There has been a significant amount of work done in the area of speaker verification recently, many of which are seen to be new and relevant. From 2010 to the present day, quite a few scientific papers and articles related to verification have been written. And the most popular of them can be considered [9] the work that introduces all the parameters and metrics used during the experiment. In this project, 3D CNN was used to train the model with DNN, and the WVU-Multimodal 2013 dataset was used, which is considered an audio set from 1083 different speakers during 4 interview sessions. The result of this work was that they achieved an average accuracy of 87.3%. However, despite the good result, it is possible to achieve better results using other techniques and architecture. As an example, in work [10], the G2E2 loss function was used, which increased the accuracy compared to their previous work T2E2 from 4.13 to 3.55 of the EER indicator, changing by 10%. EER here means a statistic that exhibits a biometric characteristic when the false acceptance rate equals to the false rejection rate (Fig. 1.3).

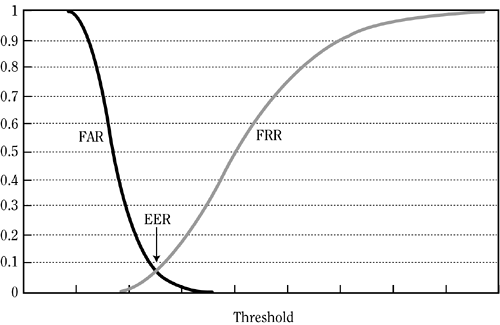


Figure 1.3 – EER

Also, without departing from the topic of the loss function, we considered it right to consider another work [11] that uses three different ones: triplet, n-pair and angular functions achieved high accuracy with a result of 3.48% EER.

However, also taking into account the work [12] where the use of eye-vector and PLDA with the SoftMax loss function was mentioned, they achieved a result of 2.28% EER on the google speech command dataset, which was collected as anonymous user queries in google.

Also, when working, it is important to take into account the properties of audio files that can store unnecessary characteristics such as noise. Therefore, it is considered appropriate to process these noises. Since many audio datasets with people's voices have noise and some of them are recorded as "in the wild". However, there are special datasets like Librispeach, which stores audio trays recorded in special audio recording studios without unnecessary noise. And at this stage, when used without noisy datasets, the trained model can give a good result, but nevertheless, the quality of our program is important for our work, so we decided at best to use datasets with and without noise to achieve good results during training and the number false predictions were minimal.

1.5 Neural Networks

According to the literature review, there are many ways to achieve good metrics in speaker recognition. Many architectures were considered, and the subcategory of neural networks was studied in detail. In our project, there are three concepts of neural networks that will be considered and used as the basis, such as CNN, d-vector cosine distance computation, and DNN.

A neural network is a system consisting of neurons, usually divided into several layers connected to each other. Such a system can be used to solve various statistical problems, but we are interested here only in the verification problem. In this case, the network computes a score for each class from the input data. The class assigned to the input object corresponds to the one with the highest score. Each layer receives the input data and returns it transformed. To do this, it calculates a linear combination, and then possibly applies a non-linear function called an activation function. The coefficients of the linear combination determine the parameters of the layer. A neural network is constructed by superimposing layers: the output of one layer corresponds to the input of the next. This superposition of layers determines the final output of the network as the result of a differentiable input function. The logistic or softmax function is used as the activation function in the last layer, which generates the final probabilities. The loss function is associated with the last layer to calculate the classification error. Usually it is cross-entropy. The values of the layer weights are learned by back propagation of the gradient: gradually for each layer, starting from the end of the network, the parameters minimizing the regularized loss function are calculated. The optimization is performed with a stochastic gradient descent. There is architecture of NN shown below (Fig. 1.4).

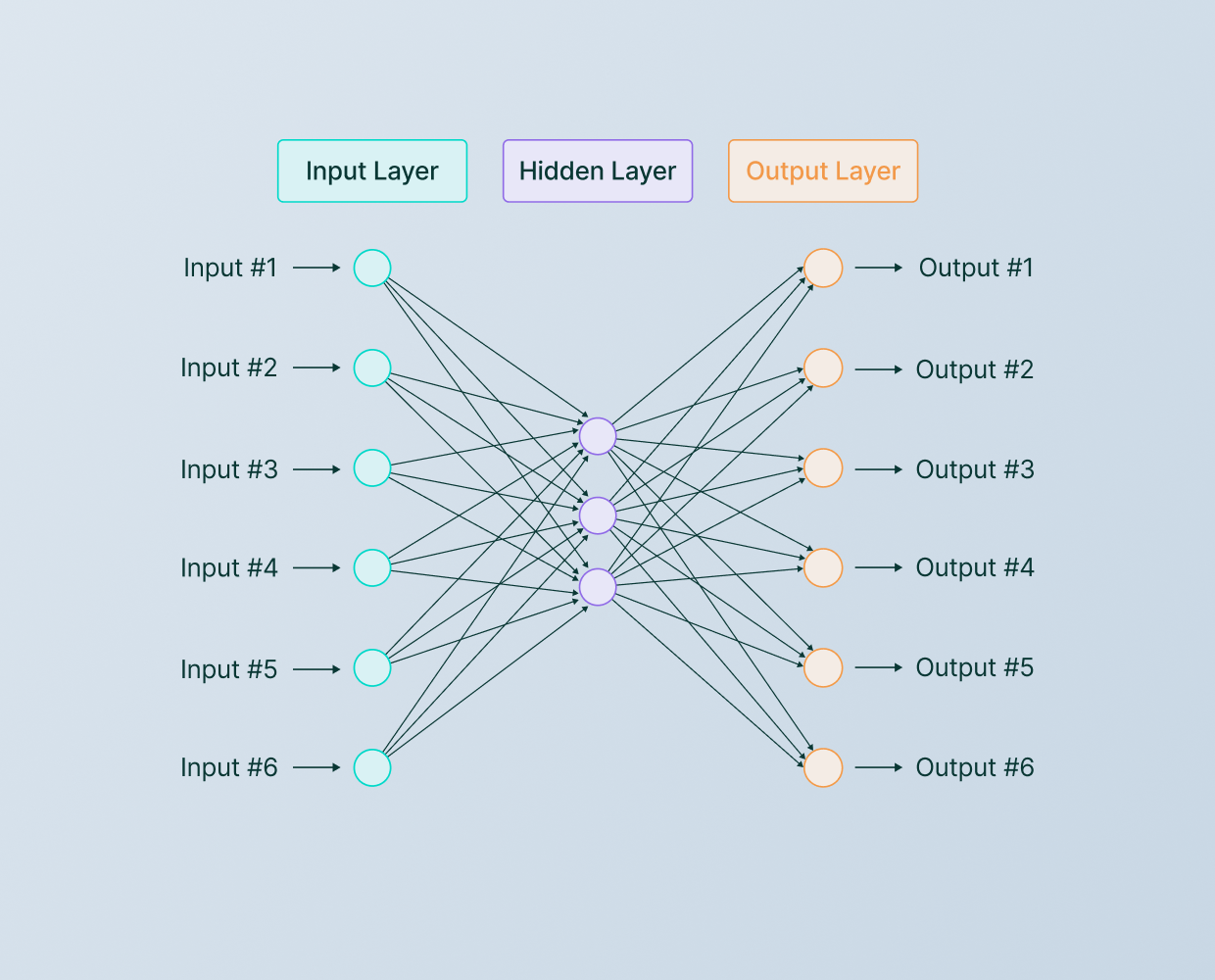


Figure 1.4 – Architecture of NN

Let us focus on the convolutional neural network, one of the most powerful deep learning algorithms. It belongs to the subcategory of neural networks: therefore, they have all the above characteristics. Convolutional Neural Network is especially often used in the field of image and sound processing (Fig. 1.5). Such a convolutional neural network usually consists of at least 5 layers. Pattern recognition is performed within each of these layers. Each layer refines the pattern recognition based on the output of the previous layer. You can think of this procedure as a small recognition grid, which moves step by step through the area of the data set being analyzed.

The convolutional neural network processes the input data as a matrix. The CNN consists of filters called convolutional layers and aggregation layers called merging layers, which are repeated one by one and at the end of one or more layers of conventional fully connected neurons.

In most cases, the results of each layer of the convolutional neural network are activated by the so-called ReLU function. This ReLU function ensures that all values less than zero become zero, and all values greater than zero are kept at a 1:1 ratio. In the case of classification tasks, the last layer gets softmax activation, i.e., the output of all output neurons is 1 in total and indicates the probability of the corresponding output.

The weights of filters and full-link layer are chosen randomly in the beginning and then continuously optimized by backpropagation during training. In the case of classification tasks, categorical cross-entropy is used to measure the error. This is the negative natural logarithm of the calculated probability for the category.

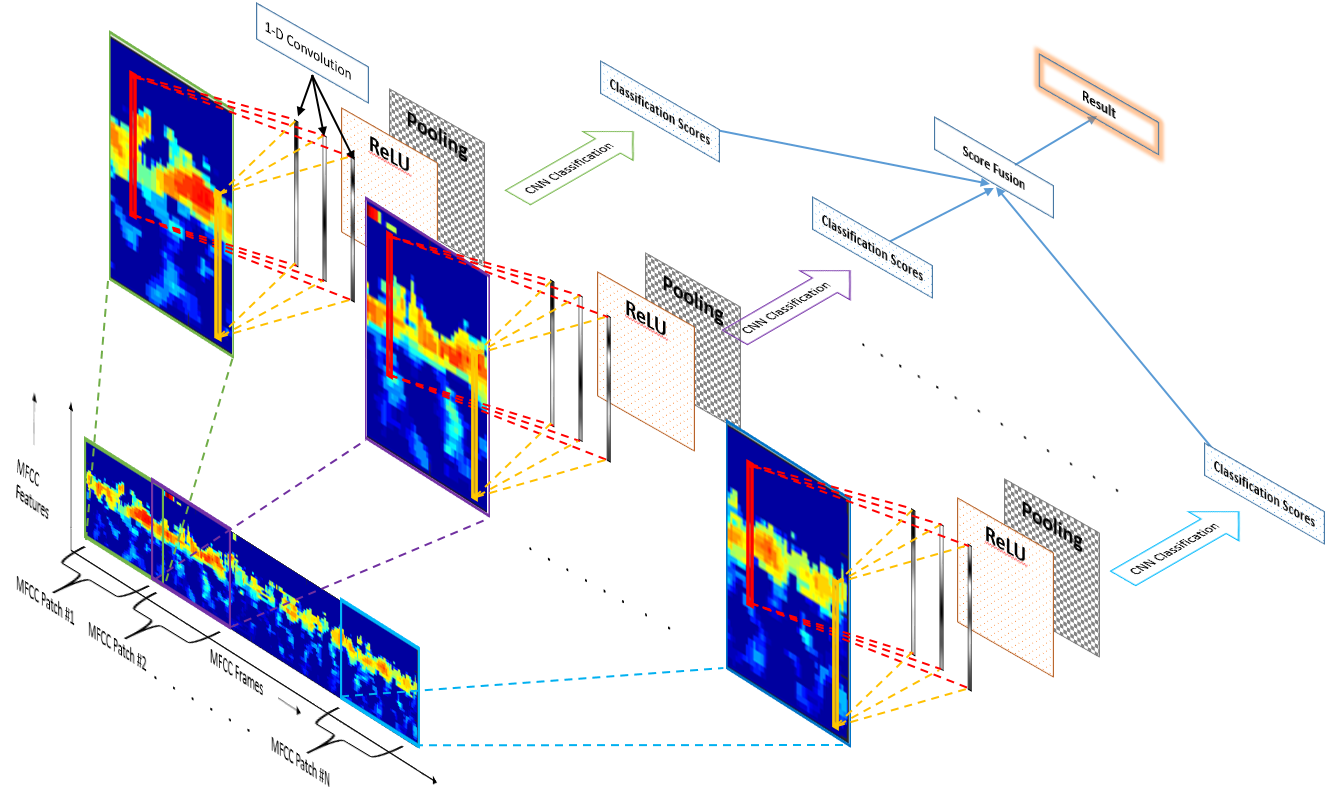


Figure 1.5 – Architecture of CNN

1.6 Team members roles

3 people are involved in the development of our project.

Menglibayev Zeinur and Askarbekov Aisar are data engineers. They collect and process incoming data, develop and train a machine learning algorithm.

Key responsibilities:

● Collection and processing of incoming data.

● Development and training of a machine learning model.

● Checking the machine learning algorithm*.*

Temirlan Khakhazov is a developer and MLOps project engineer. This team member writes the client logic and works out the layers of interaction between the server and the machine learning model. Moreover, he is responsible for configuring the server, and integration of the software product.

Key responsibilities:

● Setting ci/cd.

● Development of server and client logic.

● Code debugging.

1.7 Development Environment

Software development is done in an IDE. An IDE often includes a compiler, a tool to debug, and various other software development needs. IDEs combine several facets of computer applications. The IDE contains a sizable library for machine learning and computer science.

The IDE can be used to implement various aspects of coding such as compiling, debugging, creating executables, editing source code, and more. Python is a language that programmers often use, as well as a Python IDE for coding and compiling. There are IDEs that are widely used today. Let's take a look at Python IDEs for DS and ML on the market.

1.7.1 Jupyter Notebook

For our project, we will use two integrated development environments - Jupyter Lab (Fig. 1.6) and PyCharm (Fig. 1.7).

For data scientists and machine learning specialists, Jupiter Lab is a web-based Python programming environment. One of the key advantages is that Jupyter Lab's interactive inference engine will allow us to test our code as we develop it. Jupyter Lab has a respectable user interface because it provides a terminal, text editor, console, and file directory view all at once. It has features like auto code completion, auto formatting, autosave, etc., making it one of the best free Python IDEs for ML and DS experts. Zen mode in JupyterLab enables users to cut down on superfluous displays and distractions so they can concentrate on the current project. .py, pdf, and other file types can all be uploaded together with files developed in JupyterLab. You can also upload them as slides, such as ".png".



Figure 1.6- Home page of Jupyter Lab

1.7.2 PyCharm

PyCharm is a great python development environment that has features like auto code completion, code auto indentation, etc. It has an intelligent debugger that analyzes the code and highlights errors, so it is ideal for web development, PyCharm is preferred by many because of its ease of navigation. We can search for any specific character used in long codes using the navigation feature in PyCharm. PyCharm also makes it easier to chain multiple scripts together. With PyCharm's refactoring tool you can easily restructure your code; you may alter the method signature, rename the file, and extract any method from the code. PyCharm has a large database of frameworks that will allow you to write the server side of the application. Due to its flexibility and convenience, we will be able to connect the database, the machine learning model, and the server, thus building a web application.

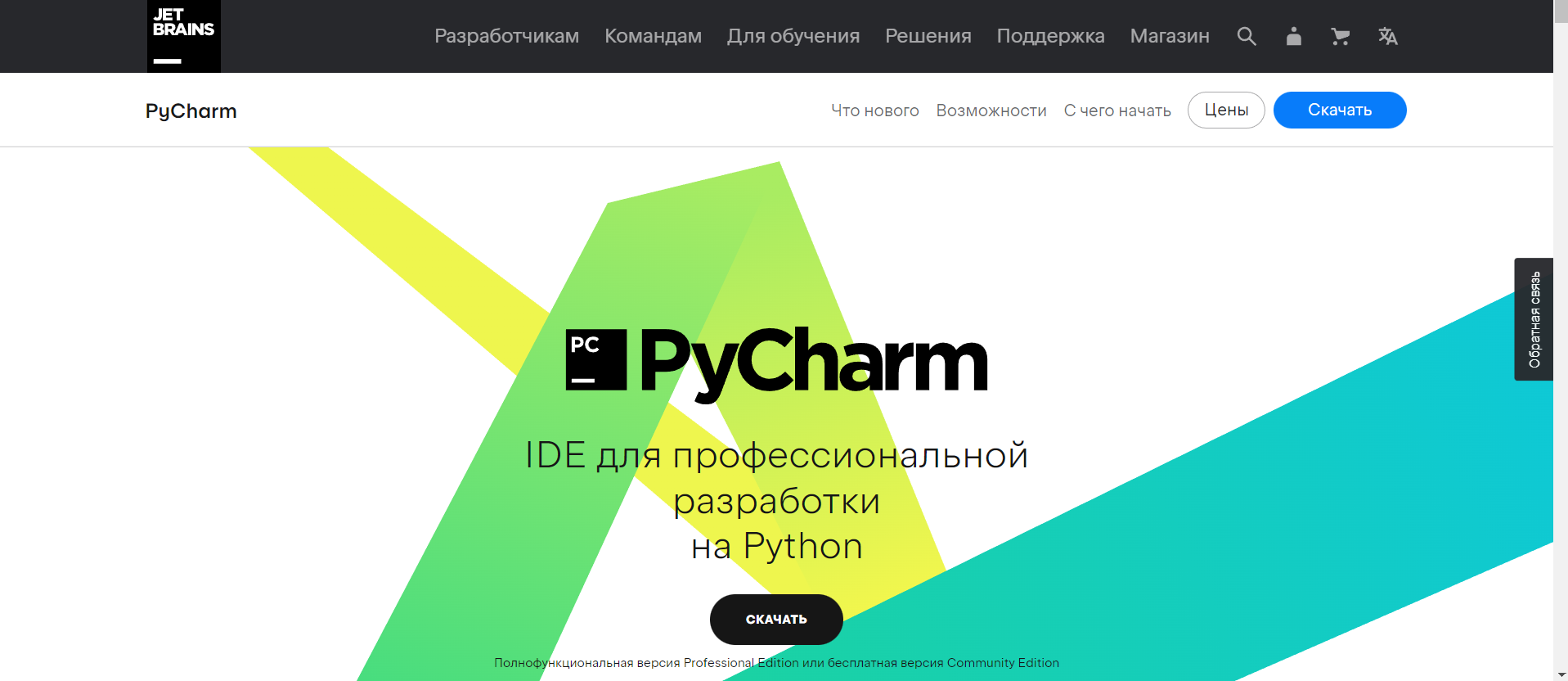


Figure 1.7- Home page of PyCharm

1.8 Programming language

There is no optimum language for machine learning, according to specialists in the field; each one is effective depending on the situation. Some programming languages are more suited to machine learning projects than others. How many machine learning engineers choose a machine learning language depends on the type of business challenge they are working on. For example, the majority of machine learning engineers prefer utilizing Python for NLP tasks, R or Python for sentiment analysis tasks, while other machine learning applications like security and threat detection are likely to be implemented in Java. Software engineers with a background in Java development may occasionally use Java as their main programming language when working with machine learning. Below is a diagram of the dynamics of the popularity of programming languages (Fig. 1.8).

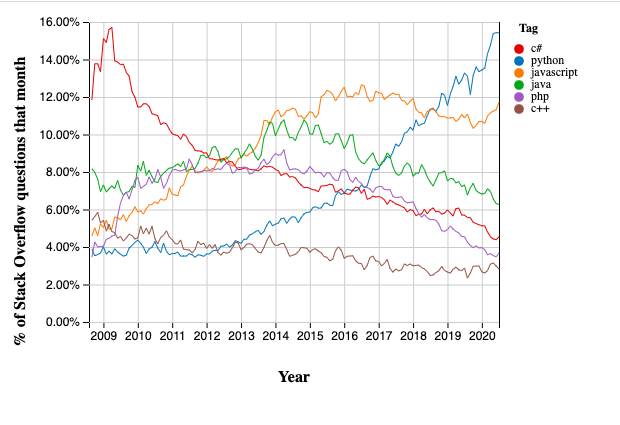


Figure 1.8- The popularity of programming languages

To implement our machine learning model, we chose python because it is ideal due to the enormous number of libraries and community, it is currently the fastest growing language in the world. Python was designed for readability and ease of use. It uses oops concepts but can also be used as a scripting language. It is preferred for natural language processing and sentimental analysis. It includes almost all packages required for machine learning. The table below shows some examples of machine learning libraries:

Substantial number of libraries and packages

As machine learning engineers, we may save time by avoiding starting from nothing by utilizing the built-in Python modules and packages. Machine learning requires continuous data processing, and Python has tools and packages built-in for every need. As a result, we machine learning engineers will observe a reduction in development time and an increase in productivity while dealing with complex machine learning applications. The wonderful thing about these tools and possibilities of packages is that there is no learning curve and that you may use them as soon as you understand the fundamentals of Python programming. Here are the reasons why this programming language was chosen for this project:

* To work with text data - we can use NLTK, SciKit and NumPy
* To work with images – we may use SciKit and OpenCV
* To work with sound - we may use Librosa
* To implement deep learning - we can use TensorFlow, Keras, PyTorch
* To implement basic machine learning algorithms - we can use SciKit - learn.
* For scientific computing - we use SciPy
* For clear data visualization, we use Matplotlib, SciKit and Seaborn.

Coding clarity

To successfully integrate sophisticated machine learning algorithms and multipurpose workflows, code readability is essential. Because of Python's straightforward syntax and emphasis on code readability, machine learning engineers may concentrate on what to write rather than how to write it. It is simpler for machine learning practitioners to discuss concepts, methods, and tools with their colleagues when the code is easy to understand. Among data scientists, Python is one of the most popular programming languages.

Flexibility

Python's multi-paradigm flexibility and support for imperative, functional, procedural, and object-oriented programming styles, machine learning developers have the opportunity to select the method that best solves their issues. Python's versatility enables machine learning specialists to select the best programming approach for a given problem. For example, although sometimes it might be beneficial to record the state of an object, other times the issue can call for sending functions as inputs. Python decreases the likelihood of mistakes while allowing flexibility in selecting any of the methods. Given that machine learning practitioners do not need to recompile source code to view changes, Python has a lot to offer in terms of flexibility when it comes to implementing changes, not just in terms of programming styles.

Versatility

Python is a flexible language that can be used to create custom APIs. Flask is one of the most well-liked frameworks for backend development. As a result, Flask offers you the technologies, tools, and libraries you need to create online apps. This web application may expand to include numerous web pages, a blog or wiki, a web calendar, or a business website. Flask belongs to the category of microframeworks. Microframeworks are, as a rule, frameworks with little or no dependency on external libraries. This has its pros and cons. The pros are that the framework is lightweight, there are a few dependencies to update and track down security bugs, the cons are that you end up having to do more work yourself or expand the list of dependencies by adding plugins.

1.9 Teamwork tools

It is crucial to introduce GitHub right away (Fig.1.9). Like a code cloud, GitHub is a service that hosts open-source repositories. It maintains track of the numerous changes made with each iteration and hosts your source code projects in a variety of programming languages. In other words, it functions as a platform for the interchange, dissemination, and storage of source code. A well-known website for hosting IT projects and team development is GitHub. Here, repositories are joined, remarks are made, and cloud storage is used on the platform. Teamwork is streamlined by the usage of GitHub.

Изображение выглядит как текст

Автоматически созданное описание

Figure 1.9- Home page of GitHub

This tool was chosen for this project for the following reasons:

* A social network for developers. Due to all the benefits it provides to developers, GitHub has been a tremendous success. It is also the world's largest programming community and may be viewed as a social network for programmers. The public sharing of projects by developers allows them to receive not only assistance but also vital information that may be of great utility. Every programmer and other enthusiast in this community has access to review projects as soon as they are published on GitHub. By doing this, any issues that the project's creator might not have spotted on their own can be brought to their attention. Even better, the public may provide him direct suggestions, saving him valuable time in the process.
* Full change traceability. On GitHub, every modification to the project is recorded in a "change log." As a result, pinpointing the precise changes made to each new edition is simple. Looking back and recognizing modifications made by a contributor is made much easier by this feature. To determine what modifications were made, who made them, and when they were done, you might look back to when the idea was first conceived.
* Open-Source Platform. Projects are displayed on GitHub as Open-Source code. Anyone may study the code and make suggestions for modifications because of this. Due to the greater flexibility of open-source projects, this is a major strength. Indeed, they can respond more quickly and change to meet market demands. On the other hand, closed-source software needs to sell its worth to the intended audience. Programmers from all around the world may collaborate on GitHub to continuously find answers to challenges in the real world. And the general people may be directly provided these options.
* Seamless collaboration. Even if they are in separate places, when several individuals are working on a project at once, there is a very good chance that they won't coordinate well or won't respect each other's workspace. As an illustration, one team member could come up with a solution to an issue that is incompatible with the strategy used by another. This issue is resolved with the help of GitHub and its version control system. Without interrupting, coworkers may do their tasks. Projects may be handled to their fullest potential in accordance with the demands of the company or organization since everyone can see and understand what others are doing in real time.

To sum up, in our project's power comes in part from its online open-source community, which assembles a team of developers, including web developers, mobile app developers, etc. Despite all of this, GitHub is still quite appealing since it can be utilized in a variety of ways without requiring you to own the code or be a developer.

1.9.1 - Google Colaboratory

It is essential to introduce the next teamwork tool that we will use to write machine learning and deep learning model is Google Colaboratory (Fig. 1.10). Google Colaboratory, often known as Colab, is a straightforward and cost-free Google service that you may use to learn about deep learning or work with friends on data science projects. This notebook allows a Python software developer to write and execute any Python program code using just a web browser. Unrestricted use of the GPU is a characteristic of Colab that distinguishes it from competing services.

Google Colaboratory employs the word "collaboration" as its name suggests. Colab makes use of the same teamwork tools. With the demands of Python newcomers as well as machine learning programmers, big data analysts, data scientists, and researchers in artificial intelligence in mind, Google developed this cloud-based Python code creation tool.

Изображение выглядит как текст

Автоматически созданное описание

Figure 1.10- Home page of Google Colaboratory

The following factors led to the selection of this tool for the project:

* Shared access to the notebook. Prior to Colab, the Python code notebook did not exist. Now that Colab files on your Google Drive are stored, you may create shared links for them. Share the link now with a coworker who is interested in working with you. Additionally, you may use Google Gmail to request the assistance of programmers.
* Installing a special library. Libraries that are not in code snippets and are not intended to function together can be installed using Colab.
* Preinstalled libraries. From one of the Google Colab libraries that are already installed, you may import the code snippet library that you need. These libraries include TensorFlow, Keras, TenPy, Pandas, Matplotlib, NumPy, and additional machine learning tools.
* Collaborative coding. For cooperative projects, collaborative coding is crucial. Your team achieves milestones faster as a result. Google Collaborative is the best solution if the team needs real-time cooperation in data science and machine learning projects.
* Cloud storage. Our files are stored in Google Colab using the allotted storage space on Google Drive. You may continue working in this manner from any device that has access into your Google Drive profile. Data is also stored in the cloud as a backup in case of emergencies.
* Integration with GitHub. To import and export code files quickly, link your GitHub account to Google Colab.
* Stability. Due to the runtime environment's hosting on Google's servers, relatively complicated code may be performed without the need for local installation or powerful computers.

In conclusion, the project will use Google Colab since it is free, has a small amount of resources, allows for tons of machine learning and data processing coding using the Chrome browser, and makes it simple to share a code notepad with colleagues for real-time programming.

1.10 Summary

Summing up, our project is to create a program for speaker verification based on audio data using deep and machine learning technology.

To achieve this goal, we decided to use different methods to identify the best one. Moreover, to further ensure client pleasure, a simple and clear telegram bot design was implemented. For this, many factors that play an important role in the creation of the program were taken into account:

* Various previous works on the topic were evaluated to determine the level of development of the topic.
* Development environments and tools that will allow us to write our program efficiently.
* Project architecture to achieve good quality.

Thanks to all these factors, it became clear to us how and what to do to get the best result.

2 ARCHITECTURE AND DESIGN

2.1 Risk mitigation

Any projects require compliance with certain requirements, according to which the fate of this project will be decided in the future, whether it will be able to achieve its goals and whether it will be able to satisfy the requirements of customers. And one of these requirements can be considered risk management. Thanks to this process, all future risks can be identified at an initial level and a solution for such problems can be found in advance. Although not all risks can be taken into account at the beginning and some of them may occur even at the end of the design, it is nevertheless important to identify and find solutions for many risks at the beginning, which will definitely facilitate our task in the future. For this reason, we decided to think about and write down the risks that may arise in the future. We found about fifteen risks, which are listed in the Appendix, indicating the level of risk, what it is associated with and the solution for this risk.

Thus, the possible risks were listed. So, now one of the goals is to write a program considering these risks in order to prevent them.

2.2 Software Lifecycle model

First of all, it is important to understand the meaning of application lifecycle management. It includes the people, processes and tools involved in the development of a program from its planning to its commissioning and decommissioning. Thus, by combining and organizing all these things during the life cycle of the program, we will get a quality product, with reliable performance and with facilitated maintenance. As well as risk management, life cycle management helps the manufacturer to comply with certain requirements and, at every opportunity, evaluate the system to identify problems and errors. In addition, leading software companies release daily updates for their products. Application lifecycle management helps companies achieve high efficiency and competitive advantage by accelerating workflows and enabling high-quality product deployments.

It is also important to note the stages of the software development life cycle as shown in Figure 2.1.

Изображение выглядит как текст, электроника, компакт-диск

Автоматически созданное описание

Figure 2.1 – Stages of software development

1. Obtaining and analyzing requirements. Consideration and obtaining of all business requirements.
2. Design. The system and software design according to the received requirements.
3. Implementation or coding. The developer’s primary attention is on producing the code.
4. Testing. Testing the written code for compliance with the requirements.
5. Deployment. After successful testing deployed to the clients for their use.
6. Maintenance. Maintenance of the program during its use by the client.

Currently, there are distinct types of software development life cycle models. How it was mentioned earlier, using these models, we can achieve a quality product for customers. In Table 2.1, we have indicated the most convenient and suitable models for our project.

Table 2.1 – Life cycle models

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Description** | **Advantages** | **Disadvantages** |
| V-Model | The V-model is a sort of SDLC model where the process runs sequentially in the shape of a V. The model of Verification and Validation is another name for it. Its foundation is the pairing of a testing phase with each appropriate development step. It is appropriate for our project as it requires testing at various stages to work properly. Project is short, requirements are well defined, and its definition is stable, that is why V-model is one of the suitable models. | 1.Project will be done from time to time completing each phase, thus V-Model fits these criteria.  2.Our project is small, well understandable; thus, we can write down each step at V-model  3.V-model is easy to use and to understand.  4.Since our project requires stability at the end, we think that it will be good using V-model to focus on errors and their fixing. | 1.We may have some ambiguities in future development, and at this point V-model will be complicated to use properly  2.This model does not support iteration of phases, which means we cannot work at one phase for a long time.  3.This model requires simple but well-defined projects. However, we are working on such a huge project for the first time, so it will not be well defined. |
| Iterative | The detailed specification should not be the starting point for an iterative SDLC model. Instead, it starts by describing and building a small portion of the software, which can be examined to find out what further is needed. Then, an updated version of the program is created for every cycle of the model by repeating this procedure. We are going to start from a rough project improving by each phase. | 1.Major requirements are defined, but there are some details. Iterative model is a nice one evolving them with time.  2.At the end we will take high-level design because by iterative model we build the product and define its design that can be added later.  3.In an iterative model, we can track the defects at early stages. | 1.This model is not suitable with changing requirements that might happen  2.The problem might arise with details that were delayed, because this model can avoid such details  3.If the requirements are changed, the completion date of the project will be delayed and lead to problems. |

Table 2.1 (continued)

|  |  |  |  |
| --- | --- | --- | --- |
| Agile | The Agile SDLC model is one of the most widely used ways to creating software. It is, in a nutshell, a methodical approach to both software development and project management. The Agile philosophy can be described as a set of principles that encourage flexibility, adaptability, and communication. Since it is a hybrid of iterative and incremental processes with an emphasis on process adaptability, it fits to our project quite well | 1.It allows you to adapt to changes faster and easier.  2. Constant attention to technical perfection is the next advantage of using this software. Since our goal is to create a good model with a high percentage of accuracy of its metrics, we will improve it each time.  3. Concentrate on the software development process, its quality, rather than planning. | 1. Sometimes Agile values get in the way. For example, avoiding detailed documentation and its design  2. A difficult dimension, that is, it is difficult to predict what kind of effort will be involved. This problem will become increasingly apparent as our project gets larger and more complex |

In the end, we have chosen the agile model because it is ideal for different tasks, and for us, the agile model has fewer disadvantages compared to other models. For example, if our main task is to track the development process and update with model improvements, then the agile model provides all of this. In addition, the sphere of machine learning often uses modern methods and tools. Some of these tools will expand and new tools will appear as new development models are introduced. And in this case, an agile model will help you cope with changing circumstances and build tools in a reliable, repeatable, and predictable process. Concluding, Agile-like methods can work for machine learning by supporting better communication, understanding goals, and communicating problems.

2.3 Project requirements

When working on projects, it is important to have clear goals and vision for the project. Moreover, in order to fulfill all of goals, we set the following requirements for the project work.

2.3.1 Documentation requirements

1. Every detail for both users and developers should be covered in the documentation.
2. A thorough description of the full project must be included.
3. The description of the architecture must be in full.
4. Must provide a thorough explanation of all available machine and deep learning models in project.

2.3.2 Functional requirements

1. The user will be able to send audio tracks to the telegram bot.
2. User will be able to get speaker verification results.
3. Data Engineer must have created model using machine and deep learning methods.
4. Data Engineer must be troubleshooting every problem with the model.
5. Data Engineer will be able to keeps updating of his own model.
6. ML Ops Engineer must have upload audio data to model for training.
7. ML Ops Engineer must have connected machine and deep learning model with bot.
8. Developer must be able to be updated user experience on bot on a regular basis.
9. Developer must have time to time works with error that might happen.
10. The manager must be able to maintain the correct work of the bot.

2.3.3 Non-functional requirements

1. Verification of personality based on audio data using machine learning techniques.
2. The model should recognize the speaker with an accuracy of over 90%.
3. Ability to retrain the model on the new audio data received, followed by its verification result.
4. Model's capacity to maintain product stability throughout the full process.
5. Ease of use of the model for developers working together on a project.

2.3.4 Design requirements

1. A streamlined user interface will be offered correctly to cover their satisfaction.
2. Interface design and features should be accessible, the telegram bot should consist of a simple and clean design.
3. Improvements should be implemented continuously based on use and users.
4. It's crucial that the navigation be clear, easy to use, and uniform throughout all pages.
5. The telegram bot will need to direct users' attention and point out where the most crucial data will be stored.
6. The content of our bot can be easily understood by visitors.

2.4 Diagrams

Below are some of the UML diagrams that the project produced. It was created because, no matter where you draw a diagram, its principles would be clear to everybody who is familiar with this graphical language because UML has a single standardized syntax for representing things. Additionally, UML diagrams help with project architectural modeling by enabling us to collect important and minor components and show the structure of the program. It will be applied afterwards to create the code.

2.4.1 Class Diagram

To have a clear understanding of the software system and the relationships between object types, a class diagram was created during the project. It is referred to as a structural diagram since it consists of a number of classes, interfaces, relationships, interactions, and restrictions. Class diagrams are employed for this. We created this diagram for this assignment (Fig. 2.2).

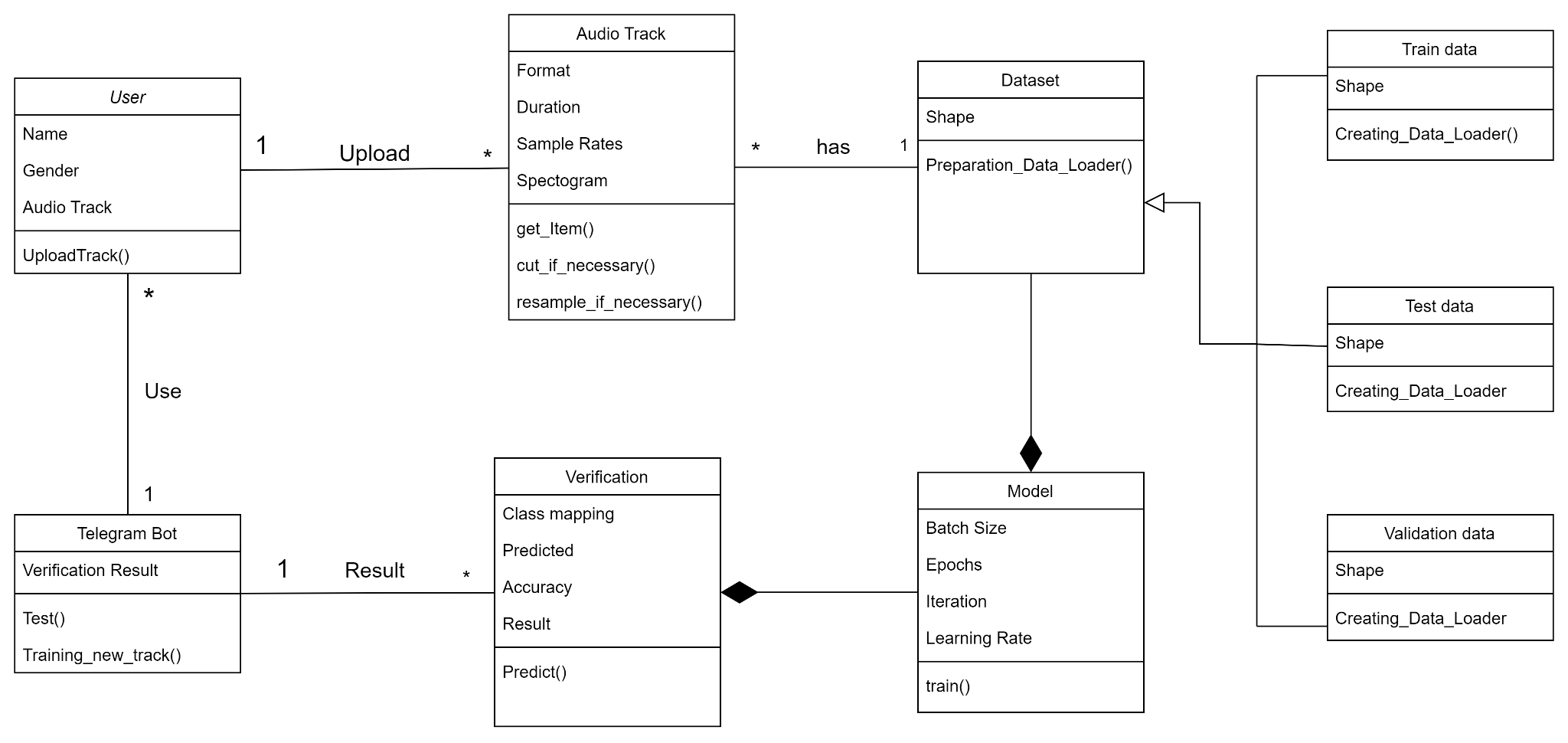


Figure 2.2 - Class diagram

We recognized the User, Machine Learning Model, and Bot main classes in our project. Additionally, classes like audio track, dataset, and verification result may be found. All classes specify characteristics, their methods, and how they interact with one another.

2.4.2 Activity Diagram

As the primary 3 classes were previously mentioned, let's see how they interact with one another at the architectural level, on the activity diagram (Fig. 2.3). An activity diagram simply depicts the flow from one activity to the next in a flow chart that describes the dynamic features of a system. A system operation is what we refer to as an activity. Control is transferred from one activity to another. This flow may be parallel, sequential, or branched. Action diagrams refer to various control flow kinds employing various elements.

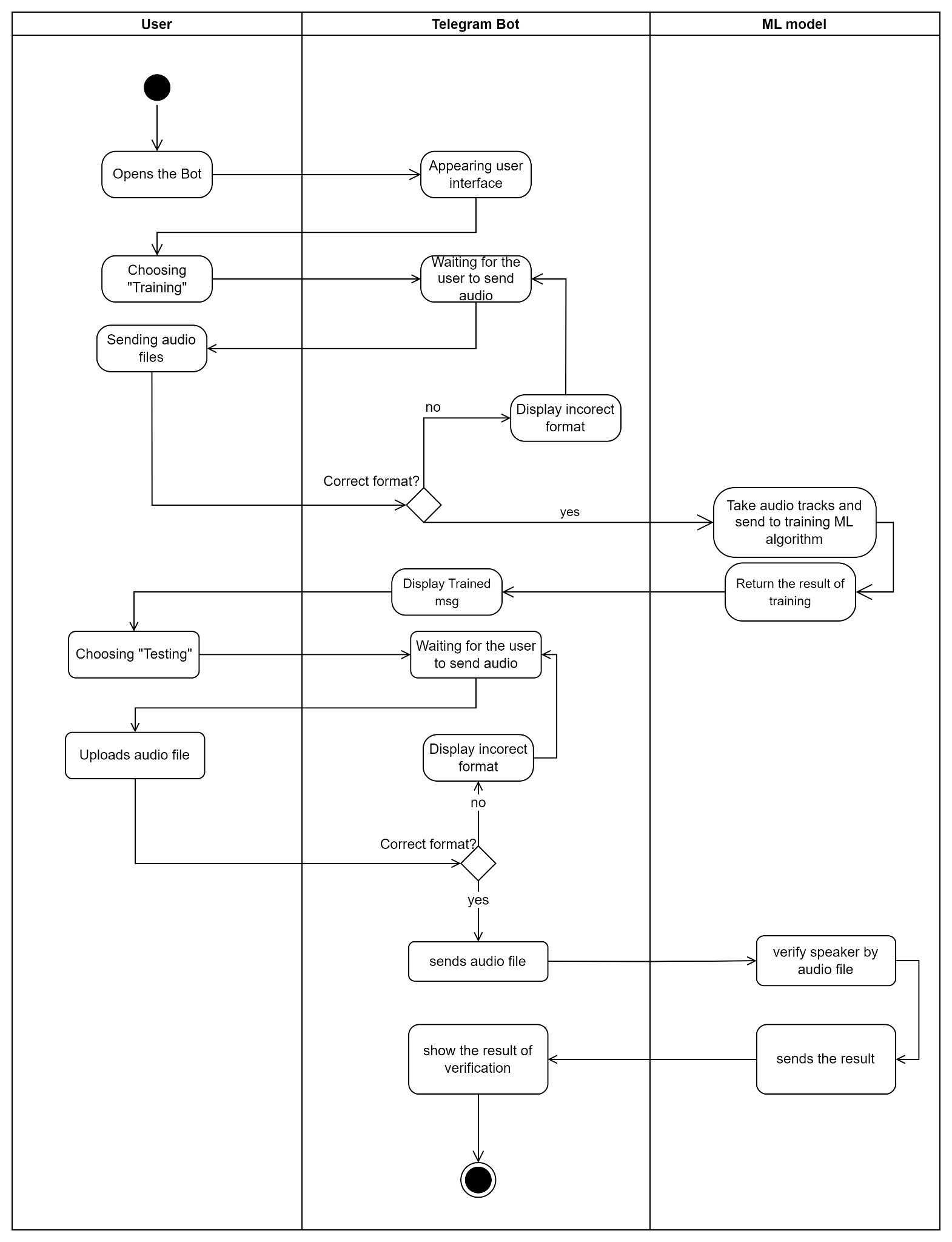


Figure 2.3- Activity diagram

Three lanes are depicted in the figure, the first of which is for users, i.e., all scenarios in which a user interacts with the bot throughout the course of this project. The second lane's goal is to display every activity that may be taken on the bot and how it communicates with other users. The third lane is a model where, upon input, audio files are received, trained on, and as a result, sent to the bot, which then displays the result to the user.

2.4.3 Sequence Diagram

It should be noted that in addition to the activity diagram, our project also features the sequence diagram listed below (Fig. 2.4). Given that it illustrates the interactions between various objects and the order in which they take place, a sequence diagram is a form of interaction diagram. The demands of a new system or the documentation of an existing process are analyzed using these diagrams by both software engineers and business managers. Occasionally, event diagrams or event scenarios will be used to refer to sequence diagrams.

We have three instances in our application: a user, a telegram bot, and a machine learning model.

Изображение выглядит как диаграмма

Автоматически созданное описание

Figure 2.4 - Sequence diagram

In summary, after entering the training page, the user sends audio data to the training, then the bot returns the page, after which the user sends data to the input of the trained model, following which the training returns the model's training results, and afterwards the bot displays these training results to the user. In this method, the sequential activity for each component, in particular the test and the outcome of the personality verifications, is described in full.

2.4.4 Component Diagram

Undoubtedly, in keeping with the architecture, it is important to note the component diagram (Fig. 2.5). The arrangement of system components and their interdependencies are represented using component diagrams. They give a high-level illustration of the parts of the system. A component diagram is used to show the connections between the various parts of a system. A module of classes known as a "component" represents separate systems or subsystems that may communicate with other parts of the system.

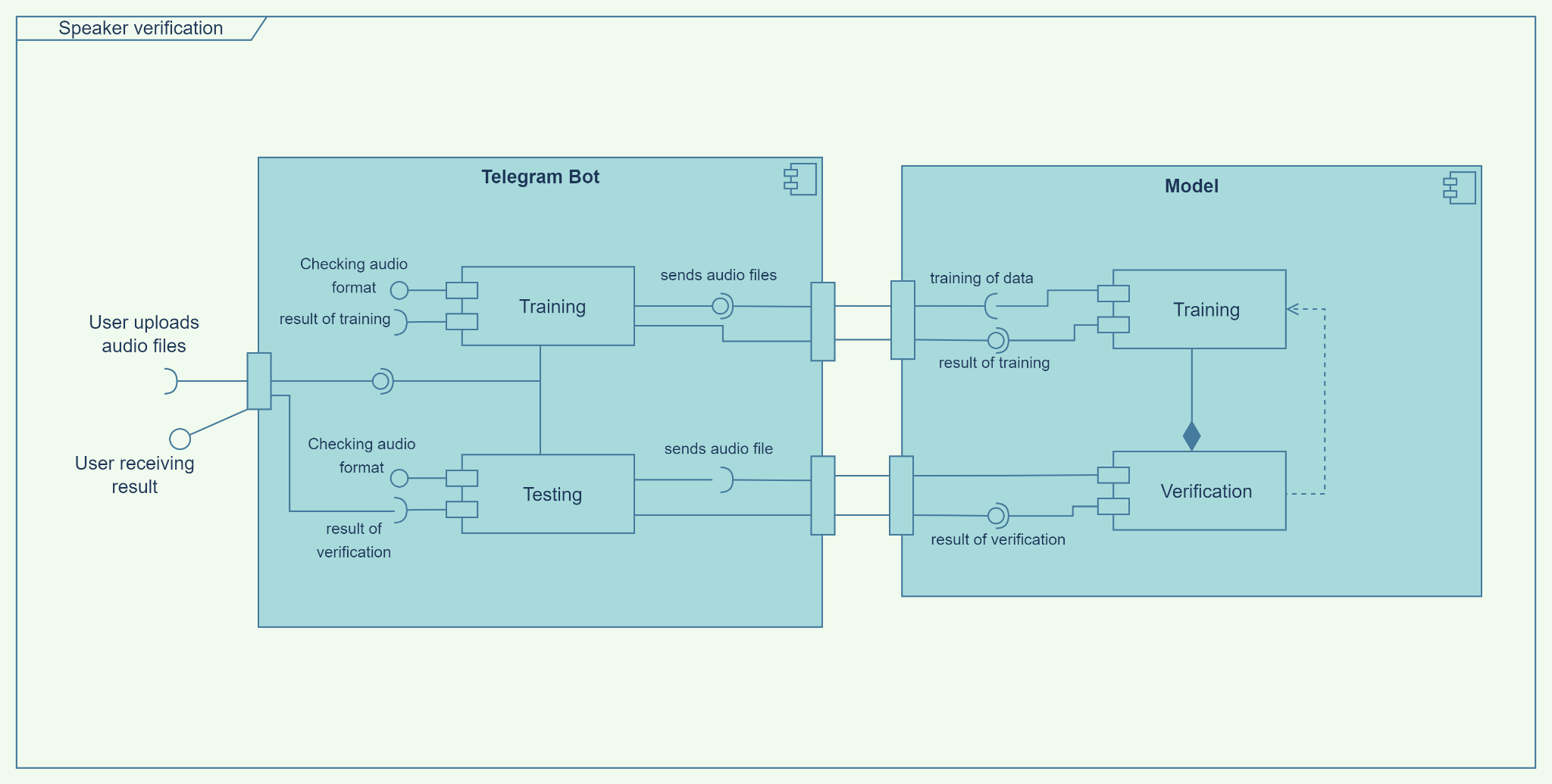


Figure 2.5 - Component diagram

According to component diagram, the bot and machine learning model are connected via interfaces, and each has a distinct role that will be carried out anytime a user wants to engage with our system.

2.4.5 Deployment Diagram

Next, it will be important to note deployment diagrams since they are used to show the system's hardware processors, nodes, and devices, as well as the communication routes between them and the placement of software files on the hardware. Deployment diagrams help in expressing the hardware architecture of the system, as compared to a variety of UML diagrams, which focus mostly on the logical elements of the system.

Deployment diagrams are often used to depict a system's physical hardware and software. It may use it to understand how the system will be physically installed on hardware.

There is a deployment diagram shown below (Fig. 2.6).

Изображение выглядит как диаграмма

Автоматически созданное описание

Figure 2.6 - Deployment diagram

We have four primary pieces of physical gear, including a browser, telegram, database server, and machine learning model, according to the project's diagram. Each piece of physical equipment has relationships and dependencies with the others. The structure's logical components, or components, as well as their interdependencies, have also been characterized.

To sum up, we have created diagrams connected to our project in order to offer a broad grasp of the architectural framework in general. We have gone through the following classifications of UML diagram such as static, dynamic, implementation. In the static UML diagram of classifications, we have provided a class diagram. The sequence diagram is part of the dynamic diagram. Finally, diagrams like component and deployment diagrams were given by implementation classification of UML.

3. IMPLEMENTATION

3.1 Use scenarios

A use scenario, sometimes referred to as a use case, describes how a system or software application would be utilized in actual settings to accomplish a particular aim or objective. It is a story that details how a user or system interacts with the application or system being utilized.

Use scenarios are crucial for the creation of software because they give developers and users a clear knowledge of how the program will be put to use in practical situations. They aid programmers with identifying potential problems and difficulties that might occur during use and guarantee that the product satisfies user requirements.

Use scenarios can be used to describe how the system or software will be utilized to analyze data and offer insights or predictions in the context of machine learning and data analysis. A use case for a machine learning model that forecasts customer churn, for instance, might outline how the model would be used to examine customer data and offer insights into which clients are most likely to leave.

However, in our case, we have a machine and deep learning speaker verification model that is deployed into a telegram bot. All actions for the user experience will be provided through the bot and all interactions between the user and model will be via the bot. The first step is to have access to Telegram, in other words, to have an account. Thereafter, the search and opening of the telegram bot will be done by typing the name of our bot or by a direct link to it (Fig. 3.1).



Figure 3.1-Initial page of telegram bot

Using our program starts with opening our bot. It will have an interface with certain commands to help the user interact with it and, at the same time, with our ML model. (Fig. 3.2)

Изображение выглядит как текст, монитор, черный, электроника

Автоматически созданное описание

Figure 3.2-Initial page of telegram bot with commands

The first thing that requires attention is the "Training" button, when it is pressed, the bot will give a message about waiting from the user's side for audio files to train his voice. Also, after downloading all the files, the bot will ask to provide the username by which it will save the characteristics of the voice.

After sending the name, the bot will send files to train the model of the new user by saving it under the sent name. After completing training, the bot will give a message about the successful training of the model. (Fig. 3.3)

Изображение выглядит как текст, монитор, сотовый телефон, экран

Автоматически созданное описание

Figure 3.3 – Training model via telegram bot

Further action with the bot will consist of a "Test" part, which the user also chooses among the commands offered by the bot. At the pressing of a button, the bot will also expect the user to send an audio file, but before that, the bot will ask for the name of the speaker to be verified.

After sending both the name and the audio file, the bot will send the data for identity verification. In this process, the model will check whether the parameters of the audio file, or more specifically the voice parameters of the recorded person, correspond to the verified identity. The bot will send a notification regarding the verification's outcome after its conclusion. (Fig. 3.4)



Figure 3.4 – Verification of speaker by telegram bot

The verification result can be either a successful verification or the speaker will not be verified.

It's also important to note that in both cases (training and testing), format of audio files will be checked during the sending part to make sure that the model will work correctly. If the format does not match the desired one, the bot will send a message about the incorrect audio file format.

In this use scenario, we have described how our program works. It does not provide for the errors that may appear in the use or before it.

3.2 Dataset

A dataset provides the machine learning model with a basis on which to learn and make predictions, making it vital for training and testing these models. When using supervised learning, the model uses a dataset of input data and associated output labels to identify patterns and generate predictions about new, unseen data. The dataset is splitted into two subsets like a training set and a testing set. By modifying the model's parameters and updating the model's weights depending on the input data and output labels, the training set is used to train the model. The model's performance and prediction accuracy on new, untested data are then assessed using the testing set.

For our purpose we decided to use a widely shared and popular dataset called “VoxCeleb”. This dataset is a large-scale speaker identification dataset designed for speaker recognition research.

The dataset consists of over 1 million from 7000 and more unique speakers, collected from YouTube videos. The speakers come from diverse backgrounds and speak in a variety of languages, making the dataset suitable for cross-lingual and cross-cultural speaker recognition research (Fig. 3.5).



Figure 3.5 – VoxCeleb dataset

The audio files in the dataset are stored in the WAV format and are sampled at 16 kHz. Each audio file is 5-10 seconds long and contains a single speaker.

The main reason for choosing this dataset is its convenience and easy accessibility. It contains enough utterance and speakers to train the model well under different conditions. Also, one of the main differences relies on "in the wild" situations. It means the audio recordings were collected from various sources, such as interviews, podcasts, and videos, with varying quality and background noise. Whereas many of the others are recorded in studio format, which means that the model can only be trained with clean speaker voices, which, when used in the real world, will reduce the accuracy of the model. Of course, there are many audio datasets similar to VoxCeleb in quality on the Internet, but they cannot boast the amount of utterance for training and further testing.

3.3 Key parts of model training and testing

To begin with, firstly we need to import all needed packages to work with them and train our model.

Next is important to note the importance of the data to be trained, therefore it is important to perform data preprocessing in which the data are cleaned, sorted and made readable not only by the user, but most importantly by the program. As a result of preprocessing the raw data will be ready for training. (Fig. 3.6).

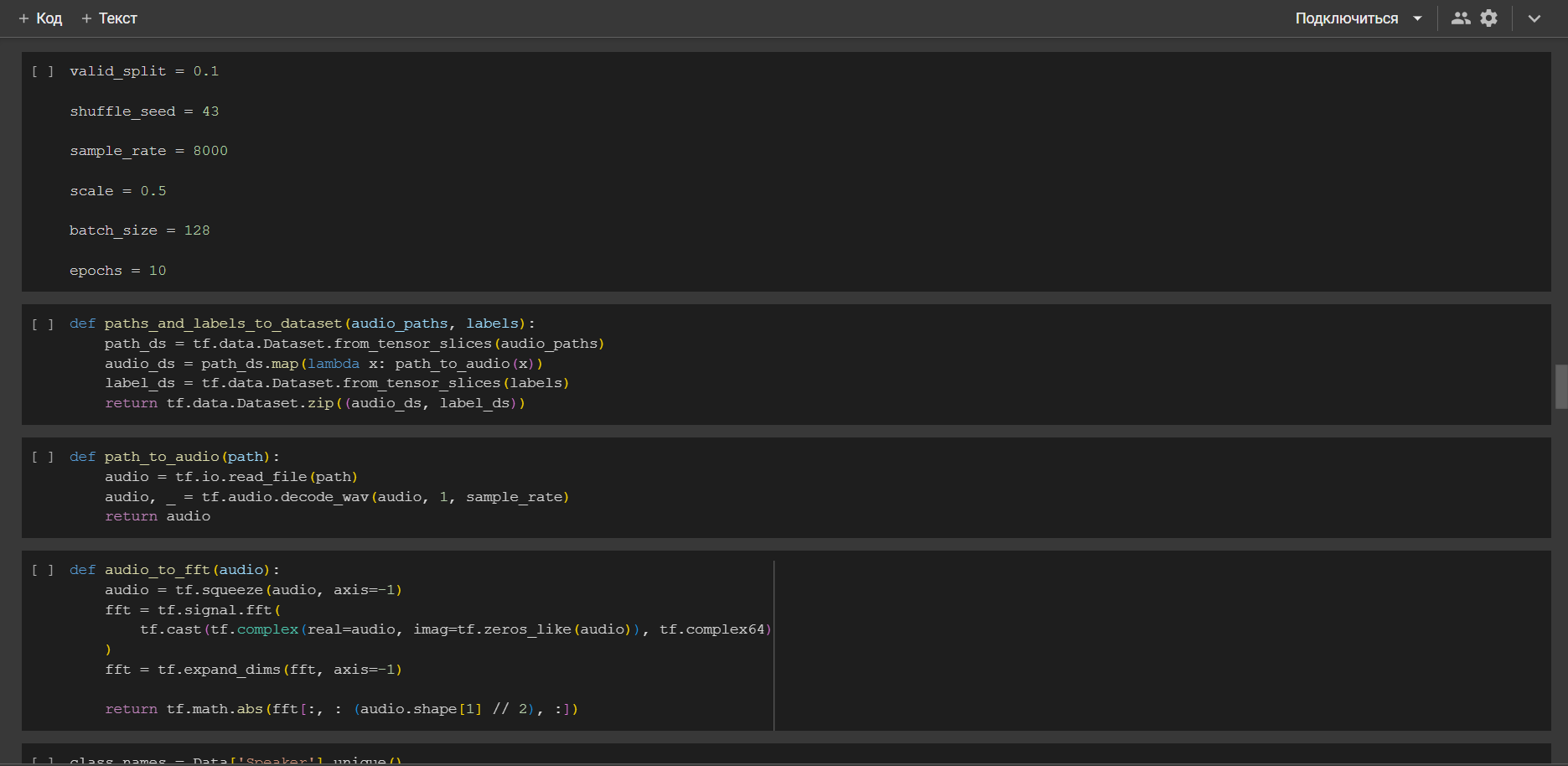


Figure 3.6 – Data preprocessing code part

The next process is to train the model under predetermined and defined parameters that will help to achieve greater accuracy. This part of the code trains the model (Fig. 3.7).

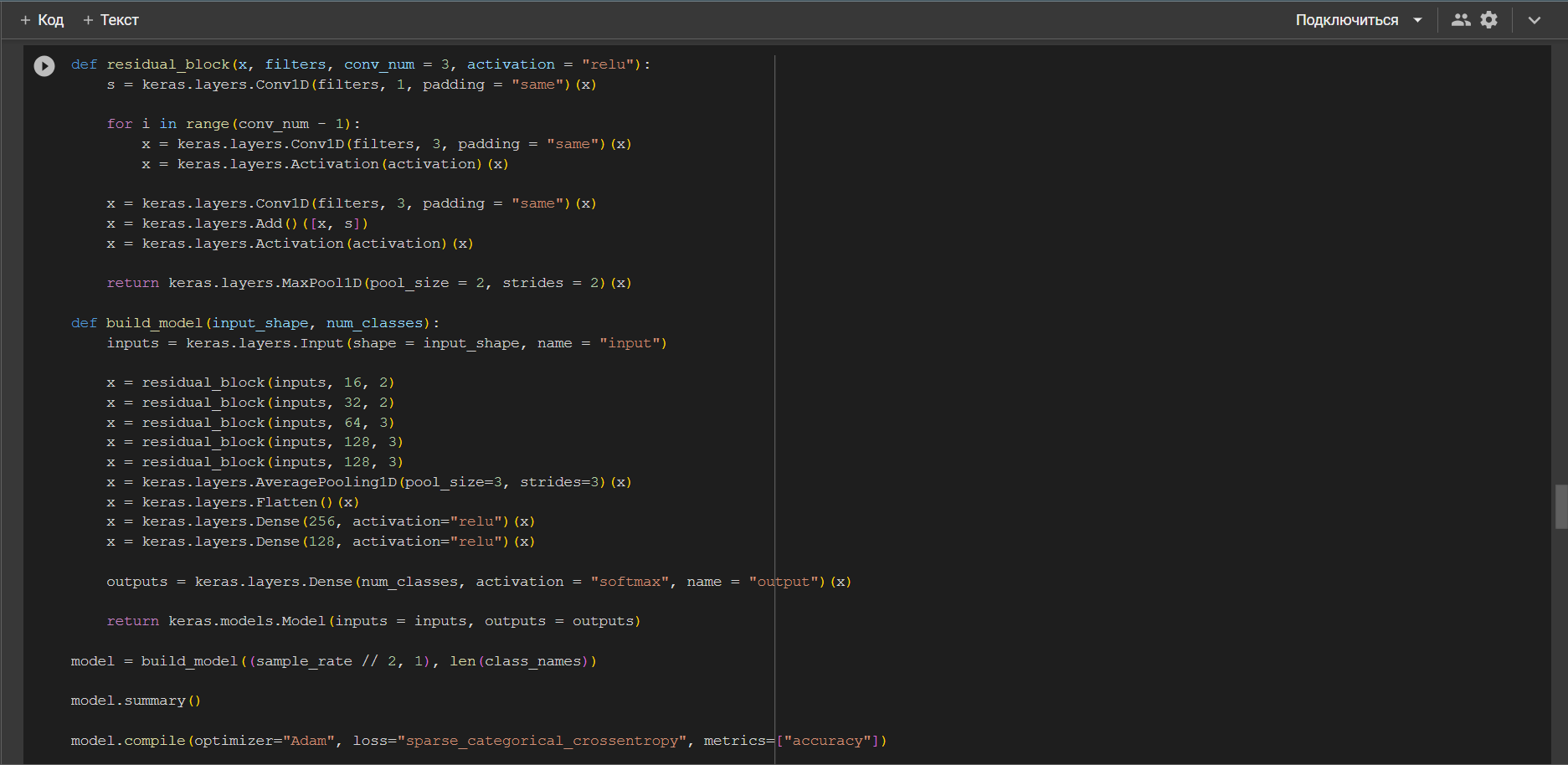


Figure 3.7 – Data training code part

And the final part is testing the trained model (Fig. 3.8).

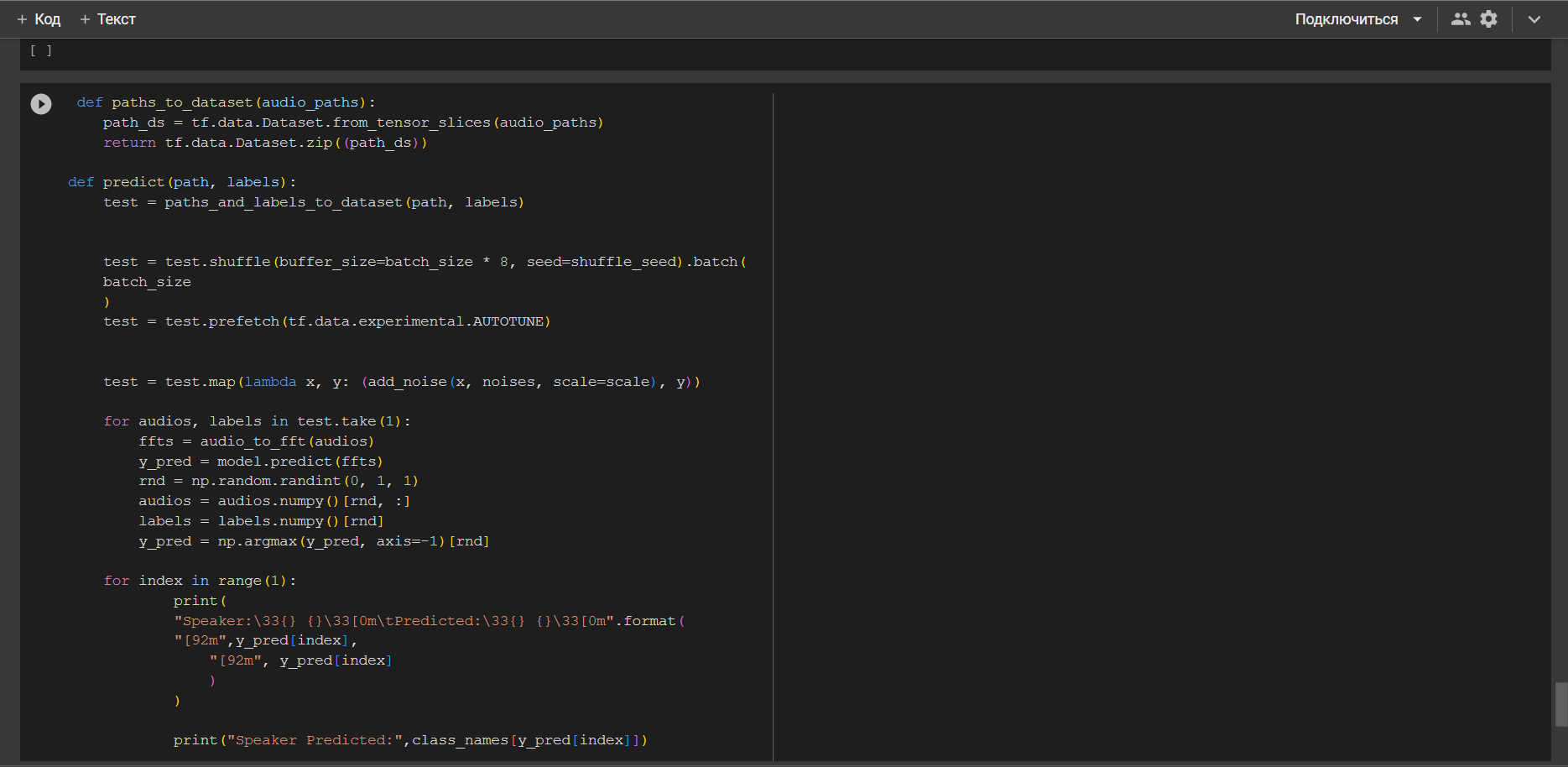


Figure 3.8 – Testing code part

Here it is also important to note that at the end of the training process and certain stages, the model is saved for use in the subsequent telegram bot.

3.4 The result

As it was already mentioned earlier, after the completion of some processes with data, such as their processing, training next is the testing process, according to the results of which we can say how well our model has trained.

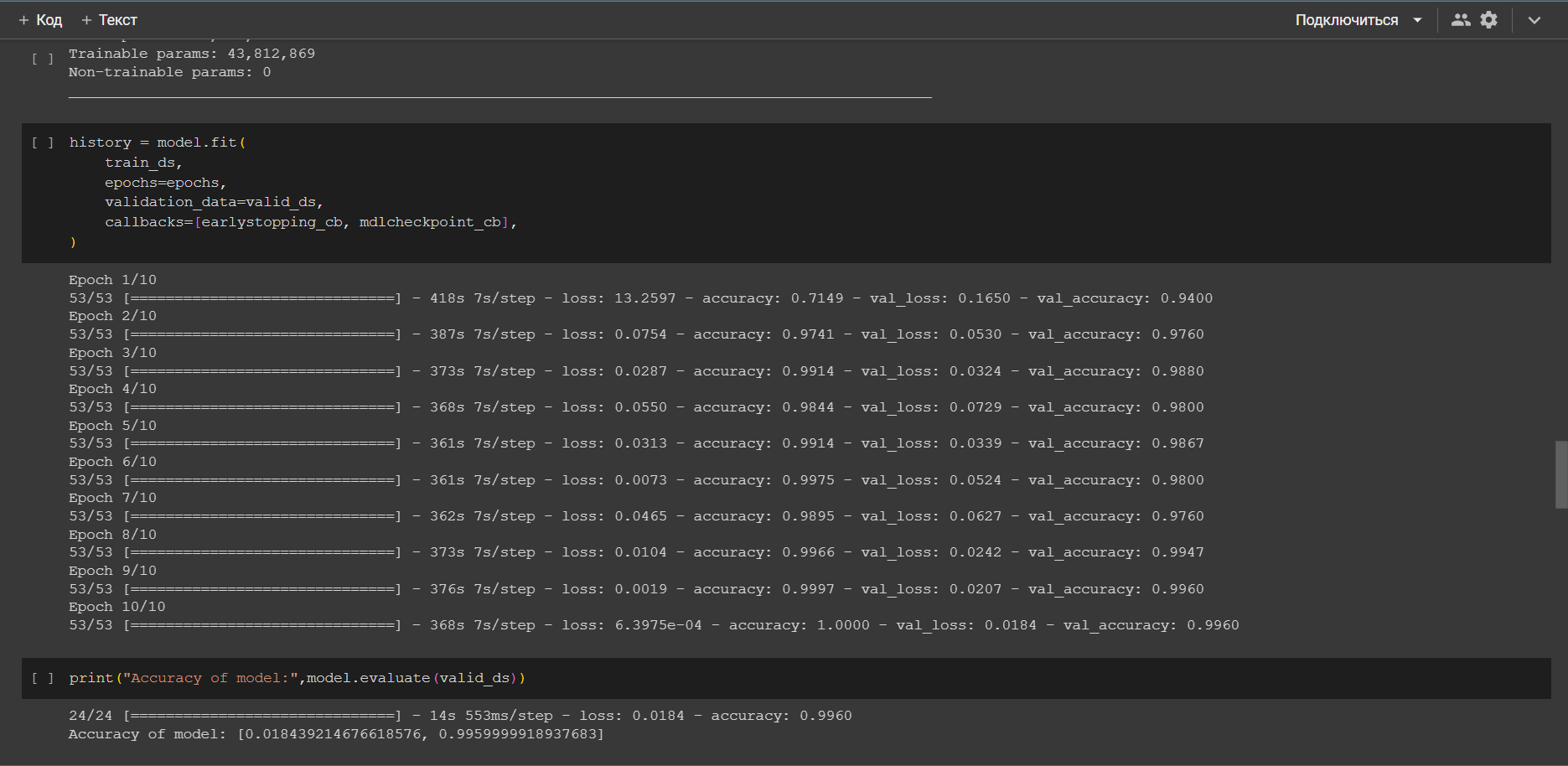


Figure 3.9 – Testing

As a result, we gained the result of loss equal to 0.02 (Fig. 3.9). Loss of 0.02 indicates that the model is working effectively at differentiating between real speakers and impostor speakers. In more detail, it means that the model correctly accepts 99% of the legitimate efforts to authenticate while accurately rejecting 1% of the impostor attempts. However, the unique use case and the system requirements determine how well the model performs.

4 ECONOMIC EFFECTIVENESS OF THE PROJECT

4.1 Technical description

The concept of our project involves the development of a machine learning and deep learning model. The purpose of which is a model of speaker verification technology, which is designed to accurately determine the identity of his voice qualities. Which will allow the use of this technology in the field of identity authentication and security systems. It can also be used in customer service applications to improve the user experience. For example, voice recognition can be used to authenticate customers and provide personalized service. It can also be used to prevent fraud in financial transactions and can also have economic benefits in healthcare. For example, voice recognition can be used to authenticate healthcare providers and patients, reducing the risk of medical identity theft.

The target group of the planned development will be companies and organizations that need such technology to verify identities and provide certain security systems.

The advantages of the services provided:

* personalization and adaptability.
* ease of use.
* safety.
* no analogue in the territory of Kazakhstan.
* saving time.
* processing in real time.

This technology allows effective verification of identities with the help of their voice characteristics and improving the quality of security.

4.2 Market analysis

The field of biometric technologies has experienced rapid development in recent years, with numerous applications in various industries. Biometric technologies refer to the use of biological characteristics, such as fingerprints, facial recognition, iris scans, and voice recognition to identify individuals. These technologies are becoming popular due to their important level of accuracy and efficiency compared to traditional identification methods.

One of the primary drivers of biometric technology development is the need for increased security in a variety of settings, including airports, banks, and government buildings. Biometric technologies can provide a prominent level of security, as each individual has a unique set of biological characteristics that can be used to verify their identity. For example, fingerprint scanning is a popular biometric technology used in many settings, including smartphones and laptops, as it provides a secure and convenient method of authentication.

Voice recognition is another biometric technology that is seeing rapid development. Voice recognition technology uses the unique characteristics of an individual's voice to identify them. This technology is used in a variety of applications, including security systems and call centers, as it provides a quick and efficient method of identification.

Nowadays, technologies based on the recognition and use of the following biometric data are actively used in the global biometric systems market:

* Fingerprints (accounting for over 50% of the total market volume).
* Face image (21.6%).
* Iris image (10.2%).
* Voice (4%).
* Vein pattern (3%).
* Palm geometry, DNA and other (about 7%).

Facial recognition technologies will also demonstrate growth rates below the market average, but the share of this technology in the global biometric systems market will grow from 21% to almost 23% (Fig. 4.1).

The fastest growing segments in the next 5-7 years will be identification technologies based on palm vein pattern, voice, and iris image [13].



Figure 4.1 – Global biometric market

Due to the rising hazards of cyberattacks using conventional identifying methods, voice biometric authentication is quickly gaining ground worldwide. Voice biometrics eliminates the need for passwords and PINs by authenticating a customer's identity and is less prone to fraud. Voice biometric services are being adopted by many industries, including telecommunication, banking and finance, healthcare, and digital commerce, for a hassle-free user experience and increased security because a person's speech is difficult to imitate. New technologies like artificial intelligence, data analytics, and machine learning are some of those driving the expansion of the global market for fraud detection and prevention systems, which are in high demand. The introduction of cutting-edge speech and voice recognition software and systems is the attention of several important players.

According to [14] the voice biometric in global market is going to reach USD 4.90 billion by 2027 and register a revenue CAGR of 23.6% over the forecast period. However, voice biometric systems have shortcomings that could limit future market revenue growth, including poor accuracy and misinterpretation, background noise interference, technical issues with accents, speech recognition, and physical side effects.

The situation in Kazakhstan with biometrics is of course different from the rest of the world, but nevertheless we are approaching the same technology that is used in other countries from fingerprint recognition to the already well implemented facial recognition. And in line with global trends, the corporate sector, primarily the financial sector, was one of the first in Kazakhstan to implement biometrics. Biometrics is present practically in every bank which works with retail segment.

Biometric data are already being used to obtain government services remotely in the eGov mobile app. But this only concerns facial recognition and speaker verification. Things are not so smooth with it in Kazakhstan. Because it is considered dynamic biometrics, in which the human voice can change depending on the situation, not all companies see this technology as useful. But as foreign experience and the trend of this technology shows, it may not be equivalent to facial recognition, but nevertheless the market share will be for this program. So, the speaker verification technology is new in our country that will gain power soon.

4.3 Marketing action plan

Any project that wants to be successful must have marketing. As important as the technological innovation employed in the Speaker Verification project is its marketing plan. The objective is to develop a marketing plan that will assist in reaching the correct audience, raising awareness, and producing leads.

That is why we need some specific roles such as:

1. Data Engineer: Employee(s).
2. Project Manager: Employee(s).
3. Back-end developer: Employee(s).
4. Data Scientist: Employee(s).
5. ML Ops Engineer: Employee(s).

Organization structure in our case is project-oriented. Reasons for choosing this type are:

* Authority is centralized.
* Unification of command.
* Each worker reports to only-one, manager.
* Project has a simple layout.
* Ability of quick decision-making.

Strategy of management:

Let us start with the project manager first. In this company, speaker verification may be completed by 1 PM. He will be in charge of managing a certain project from beginning to end. A project manager's specific duties include:

* Defining project scope, objectives, and deliverables.
* Creating project schedules, budgets, and plans.
* Recognizing and controlling project risks and problems.
* Delegating work and accountability to team members.
* Tracking the development of the project to make sure it remains on course.
* Informing team members and stakeholders on the status of the project.
* Ensuring that project deliverables adhere to high standards of quality.
* Carrying out post-project assessments to find areas that need improvement

After assigning tasks and responsibilities to team members, process will be like this:

* Collection of preexisting datasets.
* Cleaning and organization of huge and complicated databases. One Data Scientist is accountable for 1,2 steps.
* Data modeling: Data scientists create machine learning models that mine data for information and forecast future events. Together with data scientists, data engineers create scalable, effective models that may be used in real-world settings.
* Data engineers integrate the models created by data scientists with the company's systems and applications before deploying them into production.
* Data engineers make ensuring that the data pipelines and models are operating successfully and efficiently, and they also make any required improvements as needed. Three and four phases are handled by two data engineers.
* Infrastructure: To ensure that the infrastructure is properly configured to support the machine learning models, ML Ops engineers collaborate with back-end developers.
* Data management: To train machine learning models and add fresh data to the models, back-end developers give the ML Ops engineers the data they need.
* Development Framework: Back-end developers and ML Ops engineers collaborate to select the development framework that is most appropriate for the project.
* Deployment: By connecting machine learning models with the company's systems and applications, ML Ops engineers put machine learning models into use.
* Monitoring and Maintenance: ML Ops engineers keep track of how well the machine learning models are working and change them as necessary.

Testing and Debugging: To make sure the machine learning models are operating properly; ML Ops engineers and back-end developers collaborate. Two back-end developers and one ML operations engineer are in charge of the 8–12 phases.

4.4 Calculation of economic efficiency

At the initial stage, it is particularly important to analyze the main economic indicators and calculate the cost-effectiveness of the project. This gives an idea of how much financial resources are needed to implement the project, decide on the price of the product, projected revenues and allows you to calculate the payback period of the project.

In this project cost efficiency will be evaluated by equipment costs, payroll of developers and maintenance personnel, related overheads, etc.

One of the most important cost items is equipment costs. We need the following to develop our project.

Table 4.1 - Equipment costs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Name | Quantity | Cost (KZT) | Overall (KZT) |
| 1 | Laptops | 5 | 480000 | 2400000 |
| 2 | Host | 1 | 5000 | 5000 |
| 3 | Microphone | 2 | 60000 | 120000 |
| 4 | Silencer | 1 | 10000 | 10000 |
| 5 | Mouses | 5 | 3500 | 17500 |
| 6 | Monitor | 5 | 30000 | 150000 |
| 7 | HDMI cable | 5 | 2000 | 10000 |
| 8 | Overall | - | - | 2712500 |

Secondly, we need to calculate the Labor intensity and remuneration of developers. The work schedule of the web application development team is 5 working days with 2 days off (with the period of 1 and 2 working months as shown in the table 4.2).

Table 4.2 – Staff

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Specialist | Working period (m) | Quantity | Salary (KZT) | Overall Salary (KZT) |
| 1 | Data Engineer | 1 | 1 | 460000 | 460000 |
| 2 | Project Manager | 2 | 1 | 600000 | 1200000 |
| 3 | Back-end developer | 1 | 1 | 700000 | 700000 |
| 4 | Data Scientist | 2 | 1 | 500000 | 1000000 |
| 5 | ML Ops Engineer | 2 | 1 | 550000 | 1100000 |
| 6 | Overall |  |  |  | 4460000 |

Next, we need to calculate the number of mandatory deductions from wages, which includes social contributions, social tax and OSMI. Social deductions. 3.5% \* (Salary - pension contribution) = 0.035\*(4460000 - 4460000\*0,1) = 140490 KZT. OSMI. 3.5% \* Salary = 156100 KZT.

Social tax. 9.5%\*(Salary - pension contribution - contributions to health insurance from an employee) - social deductions = 0.095\*(4460000-4460000\*0,1-156100) - 140490 = 226010.5 KZT.

Thus, the payroll will be: Total salary + deductions = 4460000 + 522600.5 = 4 982 600.5 KZT.

Table 4.3 - Overheads

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Name | Quantity | Unit price (KZT) | Overall (KZT) |
| 1 | Electricity | 250kw | 17,79 /VAT | 4447.5 |
| 2 | Internet WIFI | 1 | 9900 | 9900 |
| 3 | Rent | 1 | 330000 | 330000 |
| 4 | Overall | - | - | 344347.5 |

Table 4.3 shows overheads that are not included in the direct costs of program development. So overall total cost for our program will be equipment cost + payroll + overheads = 2712500 + 4982600.5 + 344347.5 = 8 039 448 KZT.

According to our project we can make profits through:

* Advertising: the bot can be used to show targeted ads to users. If we take into account the fact of our novelty and that the flow of users bot in the beginning will be about 100, then to begin, it is advisable for us to provide space for two ads at a price of 10-20 KZT, through which we can get a certain amount of money to cover at least part of the costs. It turns out that if for each user we receive an average of 15 KZT, then a month's profits will be: the number of users per day \* advertising price \* the number of days = 100 \* 30 \* 30 = 90,000 KZT.
* Consulting and services: if the bot prove successful, there may be opportunities to offer consulting or related services to companies that want to implement similar technologies in their own projects. And in this case, we can sell our model at a price of 5000000 KZT and receive the bulk of the profits.

In the end, the total amount of profit in the first year will be given that at least one company bought our model: profits from advertising \* the number of months + model price = 90000 \* 12 + 5000000 = 6080000 KZT.

For the first year = profit/costs = (6080000-8039448)/ 8039448= -0.24.

As expected, the first year is a loss. But if we consider the profit of the second year:

1) A lot of equipment has already been purchased, of which remains host, paying ML Ops and Data Scientist staff for technical support and overheads, which amounts to 1399347.5 KZT.

2) Considering that the model was bought by at least one company and the number of users has increased to about 300 per day, the income for the second year will be (300\*30\*30) \*12 + 5000000 = 8240000 KZT.

3) Thus, the absolute economic efficiency for the second year will be (8240000-1399347.5)/1399347.5 = 4.8.

Average annual income = (-1959448+6840652.5)/2 = 2 440 602.25 KZT.

In this case, payback period = 8039448/2440602.25 = 3.29 (which is 3 years).

However, provided that our model was purchased by 2 companies in the first year, then the absolute cost-effectiveness of the project will be:

For the first year = profit/cost = (11080000-8039448)/ 8039448= 0.37(3 040 552 KZT). It will be the net profit from the first year of a project with more income than expenses.

CONCLUSION

To sum up our diploma project requires a lot of effort and costs for its implementation. Successful completion of this project will be profitable and make contribution to the development of this area. After analyzing this theme more deeply, it became clear that speaker recognition technologies can be useful in improving user experience.

Throughout the study, several ML and DL ideas, including the principles and topologies of neural networks, were investigated. To identify which algorithms and approaches would be most useful in helping us achieve our goals, we investigated and analyzed a variety of them. We developed a list of requirements that would direct our work as we created our machine learning model. These specifications covered things like model precision, training duration, and memory consumption. We were able to develop a model that matched our expectations and was effective in its implementation by outlining these needs in advance.

In the end, using a Telegram bot, we were able to effectively deploy our model and get the results we wanted. We were able to develop a more user-friendly interface that made it simpler for people to connect with our system by combining our model with a bot. This improved our system's usability and gave our consumers a more simplified experience. Our system's usability and accessibility were further improved by the incorporation of a Telegram bot, making it an efficient option for a variety of applications.

We were able to split tasks among the three of us and make good use of each person's area of expertise by working as a team. Our team's data engineer oversaw gathering, purifying, and pre-processing data, while the MLOps engineer oversaw the infrastructure and deployment of our ML models. The developer worked on putting the models into practice and incorporating them into the finished product. Despite having distinct duties, our team worked together, and everyone offered helpful criticism and support to the others. We were able to successfully work towards our shared objective thanks to the collaborative atmosphere.

Our research allowed us to carefully examine several machine learning methods and their applications. We were able to create a model that satisfied our needs and produced the intended outcomes by carefully considering our objectives and utilizing a variety of strategies and algorithms.

In our thesis, the Python programming language was used, and Jupiter Notebook, PyCharm, and Google Collaboratory were used as the development environment.

We have built possible types of diagrams to describe the process of interaction between machine learning systems. Having compared the investments and the importance of selling this technology, we made a marketing strategy and calculated the economics of our business idea.

REFERENCES

[1] A. S. Gillis, P. Loshin, “biometrics”, techtarget, 2021, [Online]. Available: <https://www.techtarget.com/searchsecurity/definition/biometrics> Accessed on 21.11.2022

[2] M. A. Godunova, “HISTORY OF BIOMETRY AND TRENDS OF ITS DEVELOPMENT IN THE MODERN WORLD”, Samara National Research University named after Academician S.P. Koroleva, Russia, Samara, st. Moscow highway, 34, 443086, [Online]. Available: <http://repo.ssau.ru/bitstream/Nauka-i-innovacii-v-XXI-veke/Istoriya-biometrii-i-tendencii-ee-razvitiya-v-sovremennom-mire-Tekst-elektronnyi-90838/1/%D0%A1%D1%82%D1%80.-156-164.pdf> Accessed on 21.11.2022

[3] Bergur Thormundsson, “Global voice recognition market size 2020 and 2026”, statista, 2022, [Online]. Available:<https://www.statista.com/statistics/1133875/global-voice-recognition-market-size/> Accessed on 25.11.2022

[4] H. Aronowitz, R. Hoory, J. W. Pelecanos, and D. Nahamoo, “New developments in voice biometrics for user authentication,” in Interspeech, Florence, Italy, Aug. 2011, pp. 17 – 20

[5] S. Imoscopi “How to Teach a Machine to Recognize Human Voices” in Kth Royal Institute Of Technology School Of Information And Communication Technology, Stockholm, Sweden 2016, pp 3-4

[6] M. M. Kabir, M. F. Mridha, J. Shin, I. Jahan, A. Q. Ohi “A Survey of Speaker Recognition:Fundamental Theories, Recognition Methods and Opportunities” in, Eds., “IEEE Access PP(99):1-1 21 May 2021,” [Online]. Available: <https://www.researchgate.net/publication/351927544_A_Survey_of_Speaker_Recognition_Fundamental_Theories_Recognition_Methods_and_Opportunities> Accessed on 03.12.2022

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

[8] G. Heigold, I. Moreno, S. Bengio, N. Shazeer, “End-to-End Text-Dependent Speaker Verification” Saarland University & DFKI, Germany 2016, pp. 1-2

[9] Wan, L., Wang, Q., Papir, A., & Moreno, I. L. (2018, April). Generalized end-to-end loss for speaker verification. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4879-4883). IEEE.

[10] Torfi, Amirsina, Jeremy Dawson, and Nasser M. Nasrabadi. "Text-independent speaker verification using 3d convolutional neural networks." 2018 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2018.

[11] Xu, Jiwei, et al. "Deep multi-metric learning for text-independent speaker verification." Neurocomputing 410 (2020): 394-400.

[12] Georg Heigold, Ignacio Moreno, Samy Bengio, and Noam Shazeer, “End-to-end text-dependent speaker verification,” in Acoustics, Speech and Signal Processing (ICASSP), 2018

[13] Central Bank, "Biometric Technologies and Trends in their Development in the World," secutek, 2021, [Online]. Available:<https://www.secuteck.ru/articles/biometricheskie-tekhnologii-i-tendencii-ih-razvitiya-v-mire> Accessed on 26.03.2023

[14] Emergen research team, "Top 8 Globally Leading Companies Offering Voice Biometrics Solutions to Strengthen Security and Authentication", emergenresearch, 2022, [Online]. Available:<https://www.emergenresearch.com/blog/top-8-globally-leading-companies-offering-voice-biometrics-solutions-to-strengthen-security-and-authentication> Accessed on 26.03.2023

[15] Campbell, W. M., Reynolds, D. A., & Singer, E. (1997). Speaker recognition: A tutorial. Proceedings of the IEEE, 85(9), 1437-1462. doi: 10.1109/5.622272

[16] Research and Markets team, “Applications of Speaker Verification Technology: Trends and Opportunities”, Research and Markets, 2019, [Online]. Available:<https://www.researchandmarkets.com/reports/5174436/applications-of-speaker-verification-technology> Accessed on 27.03.2023

APPENDIX

Table 1 – Risk Assessment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Risk** | **Importance** | **Difficulty** | **Risk level** | **Techniques for MS** | **ViewType** | **Mitigates** |
| Bad audio data quality | 5 | 5 | 5 | Architecture | Module viewtype | To mitigate this, use new equipment or analyze the audio track and improve it |
| Switching off electricity | 5 | 4 | 4,5 | Architecture | Allocation viewtype | To mitigate this risk, we can use a generator. |
| Server Down(crash) | 5 | 4 | 4,5 | Architecture | Allocation viewtype | To reduce overloading, we must install a special program that will control it. For example, if you have a lot of requests and the system gets overloaded, we must filter those requests. |
| Incorrect verification result | 5 | 4 | 4,5 | Management | Module viewtype | To mitigate Incorrect verification result we have to improve our model |
| Overtraining of the model | 5 | 3 | 4 | Management | Module viewtype | Avoid large amount of homogeneous audio data leads to overtraining of the model |
| Underfitting of the model | 5 | 3 | 4 | Management | Module viewtype | To reduce this, we have to do data augmentation, cross validation or add more data. |
| The risk of not being able to achieve satisfactory quality ML models in the allotted time | 4 | 4 | 4 | Architecture | Runtime viewtype | To mitigate we must improve our model |

Table 1 (continued)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Lack of knowledge in the subject area | 5 | 3 | 4 | Management | Module viewtype | To mitigate this improve own skills, get more knowledge |
| Does not give out identity verification results | 5 | 3 | 4 | Architecture | Runtime viewtype | To mitigate identify server issues or wrong code, then solve it by searching in internet |
| The risk that the accuracy will not be increased. | 4 | 4 | 4 | Management | Runtime viewtype | Using several types of ML model leads to improving verification accuracy |
| Non-normal distribution of data | 5 | 2 | 3,5 | Management | Module viewtype | Manually allocate speaker data, to obtain qualitative results for models |
| Error uploading audio sample | 4 | 3 | 3,5 | Architecture | Runtime viewtype | To mitigate identify server issues or wrong code, then solve it by searching in internet |
| ML models may require the use of special equipment | 3 | 4 | 3,5 | Architecture | Runtime viewtype | Try to anticipate the occurrence of this risk beforehand, and use the appropriate equipment |
| Wrong audio format | 4 | 2 | 3 | Management | Module viewtype | Put a notice on which audio extensions our site accepts |
| The quality of ML-models will not be stable over the entire period of operation | 3 | 3 | 3 | Management | Runtime viewtype | Using different types of machine learning model architectures leads to improving verification accuracy |