Ministry of Education and Science of Ukraine

ODESA NATIONAL UNIVERSITY OF TECHNOLOGY

International Competition of
Student Scientific Works

BLACK SEA SCIENCE 2023

PROCEEDINGS

ODESA, ONUT 2023

Ministry of Education and Science of Ukraine

Odesa National University of Technology

International Competition of Student Scientific Works

BLACK SEA SCIENCE 2023

Proceedings

Odesa, ONUT
2023

DETERMINING WHETHER A SENTENCE BELONGS TO A SPECIFIC LANGUAGE USING THE METHOD OF ARTIFICIAL NEURAL NETWORKS

Author: Viacheslav Mykhailov
Advisor: Oleksandr Melnykov
Donbas State Engineering Academy (Ukraine)

The problem of speech recognition and determining the language of the spoken text is considered. It was concluded that currently there are a number of applications where the method of artificial neural networks is used for language recognition. A neural network model is given. It is specified what data is received at the network input and what transformations were performed on the input data. The key characteristics that were determined experimentally are specified. The results of the model are given.

Keywords: *speech recognition, modeling, persepron, python*

I. INTRODUCTION

Today, there are a number of technical tools that are able to perceive (recognize) messages that are pronounced - computers, medical electronic equipment, cars, mobile phones, etc. There are also many commercial language recognition systems on the market: Voice Type Dictation, Voice Pilot and Viavoice from IBM; Dragon Dictate and Naturally Speaking from Nuance Communications; Voice Assist from Creative Technology; Listen for Windows from Verbex and more. These systems allow you to dictate text documents and manage your computer using voice commands. However, it should be noted that most of them work well only with spoken English [1]. It is argued that two key language recognition tasks are to achieve 100% recognition on a limited set of commands at least for one speaker and independent of the speaker recognition of continuous language flow in real time of arbitrary language with acceptable quality - despite numerous attempts to solve these tasks. over the past 50 years [1].

As a rule, the speech recognition system consists of two models: acoustic and linguistic. The computer records the sound sound in the form of a digital signal, the acoustic model is responsible for converting the language signal to a set of features that show information about the content of the language message. The program performs a complex analysis of the language, comparing audiophragms with the language samples recorded in mention. The linguistic model analyzes the information obtained from the acoustic model and forms the final result of recognition. On the basis of probabilistic calculation, the computer determines what the user could say. The model is based on the concept of phoneme - the smallest acoustic unit of language. In the learning process, the computer recognizes the most important features of the user's pronunciation and records the data obtained as a user profile [2].

Within the framework of language recognition, you can call a separate sub -face - determining the language of the text. It does not matter what the user could pronounce,

and you do not need to select individual phonemes from the said text that simplifies the solution. The development analysis showed a number of applications.

II. APPLICATION ANALYSIS

"Tools for the language of the said text" is a service that defines the audio language. Supports 8 languages (English, Spanish, Italian, French, German, Portuguese, Dutch and Russian), 3 audio formats (WAV, FLAC, OGG). The model uses coagulation and recurrent neural networks, trained for tens of hours of speech data. It is a cross -cutting model that uses the raw signal as input without making assumptions about phonetics or grammar of a language. She tries to conclude all the relevant audio recording functions based on data. The service distributes probabilities from languages that are recognized as the source. Technology can be used to classify records from 1 to 1 minute. It is stated that the accuracy of language definition increases when using longer records. For a 20-second record, the accuracy is about 95%, and for 5-second samples a little more than 80% [3].

Demov version testing showed that in the specified audio (Fig. 1) it is impossible to define language. Sentences were uttered at a moderate pace. None of the proposals was recognized. After that it was decided to use the proposed audiophragms in the languages "English" and "Italian". None of the fragments was recognized. There is no advantage of this service, since the service does not fulfill the key task - does not recognize the language.

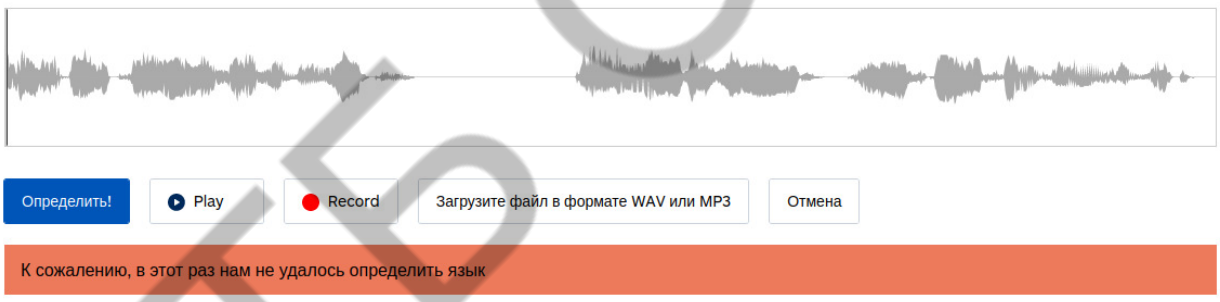

Fig. 1. Screenshot demo version «Tool Define the Language of the Text»

Speech-to-Text: Automatic Speech Recognition-a Google service that allows you to recognize language and transcribe to text. The product is implemented through API based on Google's artificial intelligence technology. Audiodans are sent to speech-to-text, in response, a text type of information sent is obtained. The service is accessed using REST API or using the language console into text. In the documentation, a separate section is devoted to language support. It contains a list of specific language recognition models that are recommended to be applied to certain languages. All of these models can only be applied to 3 languages: English (Singapore), English (United States), French (France) [3].

Demo version test showed that for accurate speech recognition (Fig. 2), it is necessary to pronounce sentences at a moderate pace and without obvious obstacles on the background. English, Ukrainian and Russian were tested. In some cases, punctuation signs were placed during the test. Ten of several proposals were recognized.

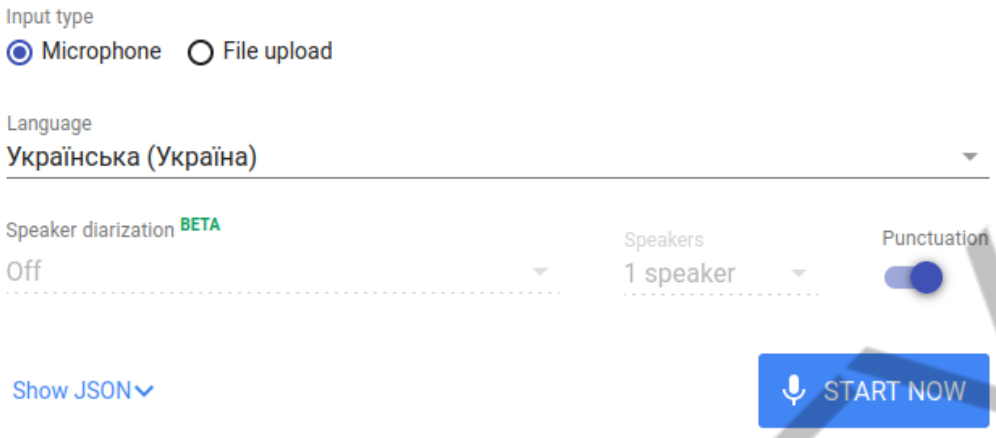

Fig. 2. Screenshot demo version «Speech-to-Text: Automatic Speech Recognition»

The advantages include the following: detailed documentation for installation and subsequent configuration of the API; clear work of the system; possibility of recognition of continuous language; the possibility of recognition of audio files; The whole pool of the languages that are supported. The disadvantages include: closed technology; high cost of product.

«Silero Service» is a service that specializes in language recognition for commercial organizations. The product provides different solutions depending on the needs of the customer. According to the author, the service supports the following languages: Russian, English, German, Spanish. The developers' website presents a demo -version with the possibility of speech recognition of both files (MP3, WAV, OGG) and streaming speech lasting no more than 20 seconds (recognizes only Russian) [3].

Demov version testing showed that the service allows you to recognize the language qualitatively (fig. 3). Only Russian can be tested in demoversion. Short sentences were uttered at a moderate pace. Ten of ten proposals were recognized.

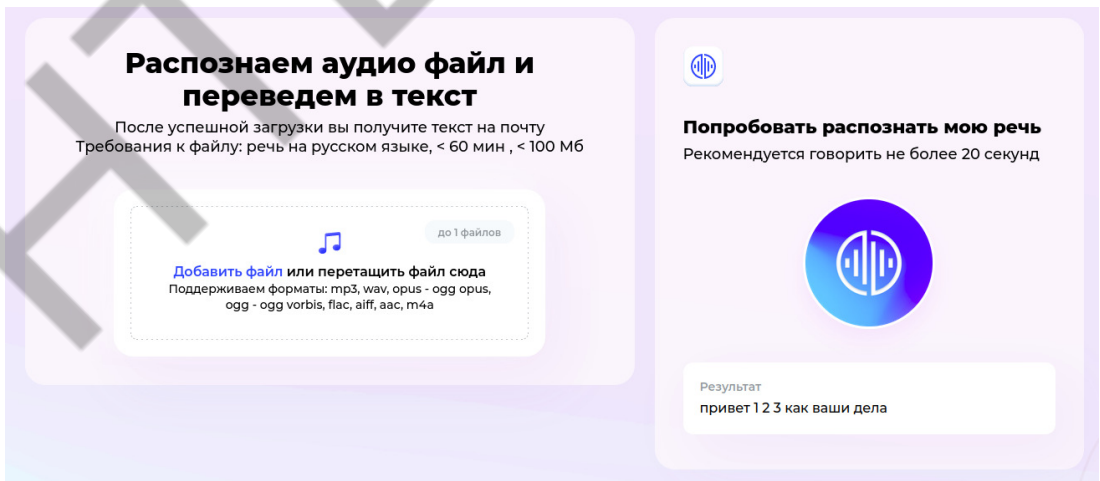

Fig. 3. creenshot demo version «SileroAi»

The benefits can be attributed to:

– the ability to use quality, open API from a commercial service to develop a solution to your task;;

– possibility of recognition of continuous language;

– The possibility of recognition of audio files.

The disadvantages can be attributed:

– closed API (partially);

– only commercial use;

– A small pool of languages.

All the examples considered to determine the language use the method of artificial neural networks. However, there is no specific architecture of the network used anywhere. All described systems are commercial solutions and have high cost.

III. OBJECT, SUBJECT, AND METHODS OF RESEARCH

The development analysis showed a number of applications. All the examples considered to determine the language use the method of artificial neural networks. However, there is no specific architecture of the network used anywhere. All described systems are commercial solutions and have high cost.

The task of developing the information system (first - decision support systems, further mobile application), which allows after the analysis of the voice message recording to determine what language this message was made.

To do this, it is advisable to use Python programming and Pywt, Soundfile, Pickle, Numpy. In the future, there is a possibility of using outsiders open API Silero Service [7].

Network architecture is a multi -layered pericеptron that contains n neurons of the input layer, three hidden layers in size (N + 2) each and the output layer in size 2 (fig. 4). The answer is given in the form of probability. The method of reversing errors using a gradient descent algorithm was used to teach a neural network. During the initialization of the neural network, neurons weight is set by randomly, followed by adjustment. Sigmoid feature was used as activation function. The error was calculated by means of an average deviation [4].

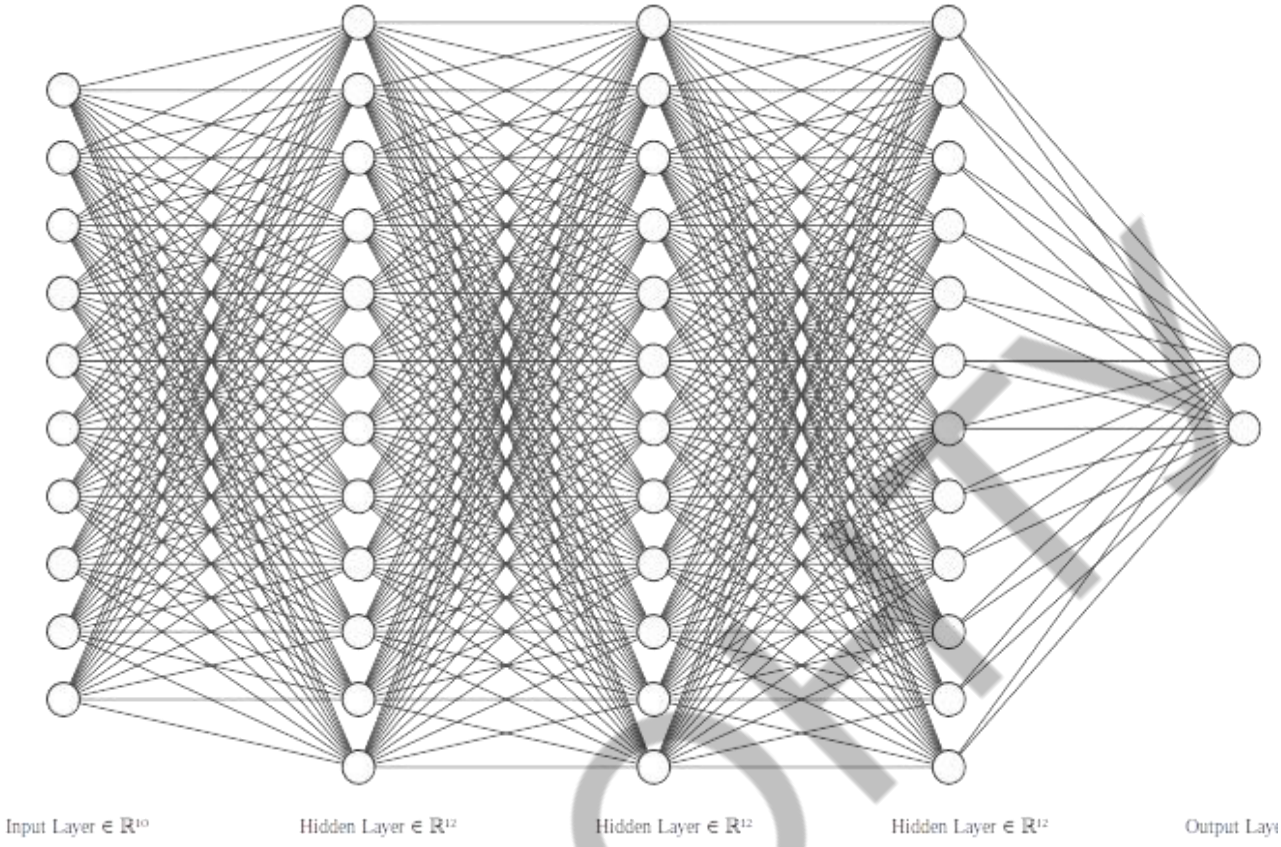

Fig. 4. Example neural network of MLP 10x12x12x12x2

The input data comes as converted discrete values. The conversion implies that discrete values with discrete wavelet transformations were thinned, resulting in the coefficients of approximation and detail. This procedure can be repeated several times. The key parameter is the decomposition level - DL [7]. In fig. 5 shows the block diagram of the values given to the appearance of the matrix.

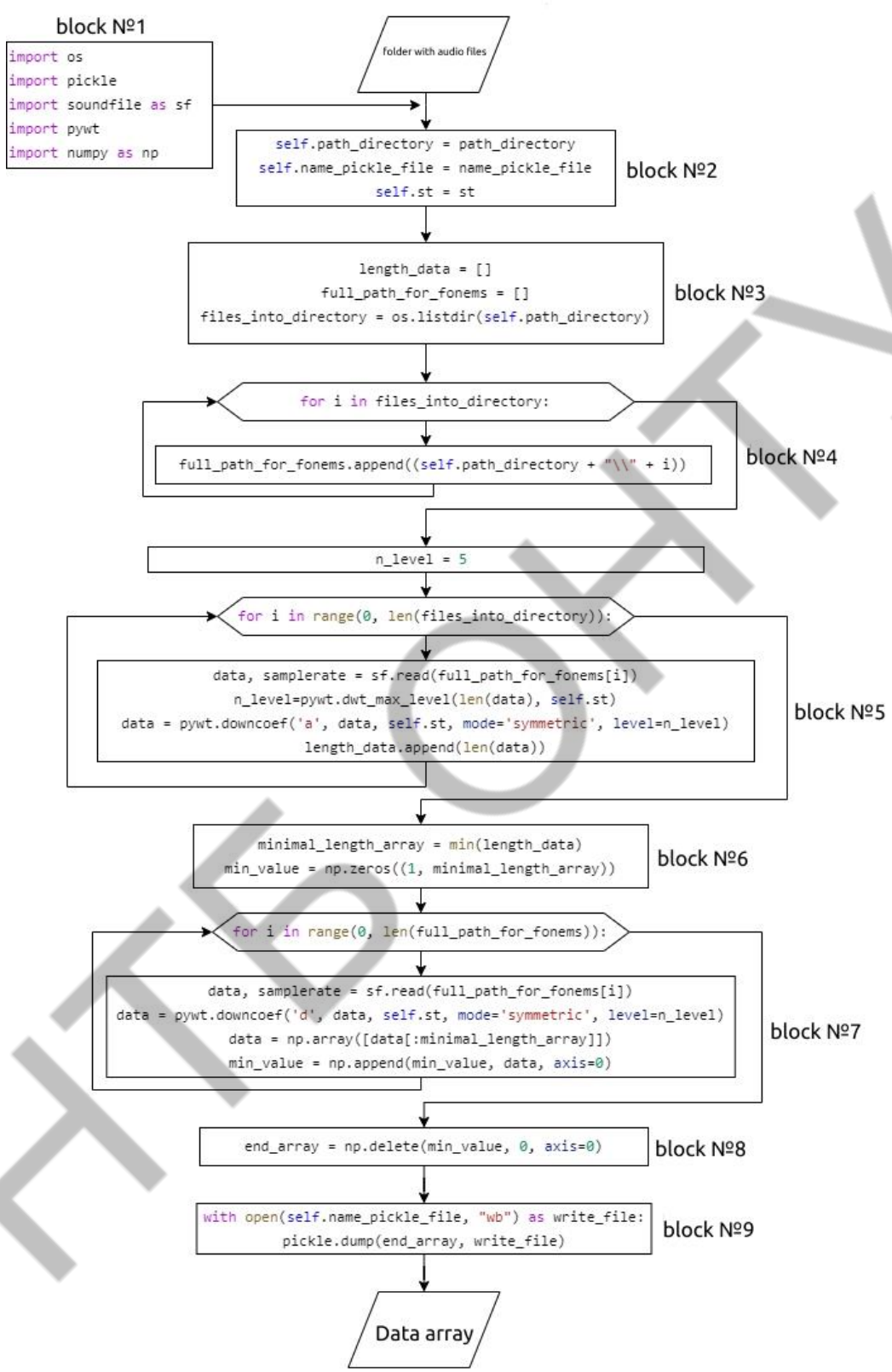

Fig. 5. Algorithm of leaving data to the appearance of the matrix of values

Block №1 - module imports. The OS module is a set of functions that allows you to operate with the operating system. In this case, it is needed to read the file names in the specified Directory. Pickle module is an algorithm of series Python objects. In this case, they are used to pack the already converted data. Soundfile module includes a huge set of features to work with WAV files. In this case, we need to read audio data from WAV files. The Pywt module has a wide range of features to work with wavelet-analysis. In this case, it is needed to obtain approximating values from the auditors obtained using the Soundfile Library. The Numpy module is an extension that allows you to support most multidimensional arrays and matrices. This module will further train a neural network, as methods built into Python will be problematic.

Block №2 - Initialization of input parameters is carried out, as well as the appropriation of a variable copy of the class. The first parameter is the way to the audio base (example: Path_directory = "D: \ Dyplom \ Code \ Test_Data"). The second parameter is the name we want to set the final file (example: Name _ Pickle_file = "Test.pickle"). The third parameter is the choice of the Weivlet family (example: ST = "DB30"). It should be noted that the output parameters of an instance of the class are assigned to a local variable class (example: self.path_directory = Path_directory)

Block №3 - An empty variable, such as a list, as well as forming a variable Files_into_directory, which contains the names of the files of the specified directory in the variable Self.path_directory.

Block №4 - the repeated path to the file in the specified directory in the instance of the class is formed.

Block №5 - reading data from an audio file. The calculation of the maximum level of decomposition is performed to calculate the approximating part. After that, on the basis of the received data, the transformation is performed to obtain the approximating part, followed by the calculation of the length of the received data. I note that this is necessary in order to calculate the required length for the entire sample, since the training of the neural network requires a fixed value of the entire matrix.

Block №6 - the minimum value is detected, as well as the creation of a zero matrix with the detected minimum value.

Block №7 - all audio data in wav format is read. A fixed value of the length of the matrix is given, based on the minimum length found among all audio files. The matrix is formed by adding values to the zero matrix specified in block №6.

Block №8 - the zero row of the row of the final matrix is deleted, because it contains zeros.

Block №9 - recording of the received matrix into a pickle format file is performed.

The recognition consists in the fact that we select the approximating parts using wavelet analysis, train the neural network using the matrix of the obtained values, and at the output we get two output neurons[6]. In that case, if it is found that it is English, the first output neuron will acquire the value of one, and the second one will acquire the value of zero. If the neural network decides that it is a Russian language, the second neuron will

acquire the value of one, therefore, the first neuron will acquire the value of zero. The process of weight adjustment is as follows:

$$HNinput_1 = W_{ti} \cdot TI \quad (1)$$

where:
$HNinput_1$ — matrix of input values of the first hidden layer;
W_{ti} — matrix of weights of the input layer TI;
TI — training data matrix[5].

$$HNoutput_1 = f_{activation}(HNinput_1) \quad (2)$$

where:
$HNoutput_1$ — matrix of output values of the first hidden layer;
$f_{activation}(HNinput_1)$ - applying the activation function (sigmoida) over the input data of the first hidden layer $HNinput_1$[5].

I note that the selection of weights is performed randomly according to the normal law of distribution in the range from [- 0.5: 0.5].

$$HNinput_2 = W_{hn1} \cdot HNoutput_1 \quad (3)$$

where:
$HNinput_2$ — matrix of input values of the second hidden layer;
W_{hn1} — matrix of weights of the first hidden layer HN_1;
$HNoutput_1$ — matrix of output values of the first hidden layer.

$$f_{activation} = \frac{1}{1+e^{-x}} \quad (4)$$

where: $f_{activation}$ — activation function (sigmoida).

$$HNoutput_2 = f_{activation}(HNinput_2) \quad (5)$$

where:
$HNoutput_2$ — matrix of output values of the second hidden layer;
$f_{activation}(HNinput_2)$ - applying the activation function (sigmoida) over the input data of the second hidden layer $HNinput_2$

$$HNinput_3 = W_{hn2} \cdot HNoutput_2 \quad (6)$$

where:
$HNinput_3$ — matrix of input values of the third hidden layer;
W_{hn2} — matrix of weights of the first hidden layer HN_2;
$HNoutput_2$ — matrix of output values of the second hidden layer.

$$HNoutput_3 = f_{activation}(HNinput_3) \quad (7)$$

where:

$HNoutput_3$ — matrix of output values of the third hidden layer;

$f_{activation}(HNinput_3)$ - applying the activation function (sigmoida) over the input data of the second hidden layer $HNinput_2$

$$ONinputs = W_{hn3} \cdot HNoutput_3 \quad (8)$$

where:

$ONinputs$ — matrix of input values of the final layer;

W_{hn3} — matrix of weights of the third hidden layer HN_3;

$HNoutput_3$ — matrix of output values of the third hidden layer.

$$ONoutputs = f_{activation}(ONinputs) \quad (9)$$

where:

$ONoutputs$ — matrix of output values of the final layer;

$f_{activation}(ONinputs)$ - applying the activation function (sigmoida) over the input values of the final layer $ONinputs$.

$$d_{output} = ONetalon - ONoutputs \quad (10)$$

where:

d_{output} — error matrix of the final layer;

$ONetalon$ — reference matrix of values of the final layer;

$ONoutputs$ — the resulting matrix of values of the final layer[5].

$$d_{hn3} = w_{hn3}.T \cdot d_{output} \quad (11)$$

where:

d_{hn3} — error matrix of the third hidden layer;

$W_{hn3}.T$ — transposed matrix of weights of the third hidden layer HN_3;

d_{output} — error matrix of the final layer.

$$d_{hn2} = w_{hn2}.T \cdot d_{hn3} \quad (12)$$

where:

d_{hn2} — error matrix of the second hidden layer;

$W_{hn2}.T$ — transposed weight matrix of the second hidden layer HN_2;

d_{hn3} — error matrix of the third hidden layer.

$$d_{hn1} = w_{hn1}.T \cdot d_{hn2} \quad (13)$$

where:

d_{hn1} — error matrix of the first hidden layer;

$W_{hn1}.T$ — transposed weight matrix of the first hidden layer HN_1;

d_{hn2} — error matrix of the second hidden layer;

$$F'_a = F_a \cdot (1 - F_a) \tag{14}$$

where:
F'_a — derivative activation function (sigmoida);
F_a — layer output values[5].

$$W'_{ti} = W_{ti} + F'_a(HNoutput_1) \cdot TI.T \cdot a \tag{15}$$

where:
W'_{ti} — updated matrix of weights of the input layer *TI*;
W_{ti} — matrix of weights of the input layer *TI*;
F'_a — derivative activation function (sigmoida);
TI.T — transposed matrix of input values TI;
α — learning rate[5].

$$W'_{hn1} = W_{hn1} + F'_a(HNoutput_2) \cdot HNoutput_1.T \cdot a \tag{16}$$

where:
W'_{hn1} — updated weight matrix of the first hidden layer *HN1*;
W_{hn1} — matrix of weights of the first hidden layer *HN1*
F'_a — derivative activation function (sigmoida);
$HNoutput_1.T$ — the transposed matrix of the output values of the first hidden layer;
α — learning rate.

$$W'_{hn2} = W_{hn2} + F'_a(HNoutput_3) \cdot HNoutput_2.T \cdot a \tag{17}$$

where:
W'_{hn2} — updated weight matrix of the second hidden layer *HN2*;
W_{hn2} — matrix of weights of the second hidden layer *HN2;*
F'_a — derivative activation function (sigmoida);
$HNoutput_2.T$ — the transposed matrix of the output values of the second hidden layer;
α — learning rate.

$$W'_{hn3} = W_{hn3} + F'_a(ONouputs) \cdot HNoutput_3.T \cdot a \tag{18}$$

where:
W'_{hn3} — updated weight matrix of the third hidden layer *HN3*;
W_{hn3} — matrix of weights of the second hidden layer *HN3;*
F'_a — derivative activation function (sigmoida);
$HNoutput_3.T$ — the transposed matrix of the output values of the third hidden layer;
α — learning rate.

IV. RESULTS

Five directories filled with samples of the Russian and English languages were created to check the functionalityspeech recognition.

Each directory counted 5 samples of each language. In the table 1 shows the results of the neural network of two directories.

Table №1 – The results of the neural network

eng	0.99979599	0.00020401
eng	0.99975415	0.00024585
eng	0.99976293	0.00023707
eng	0.99995951	4.04938865e-05
eng	0.99808374	0.00191626
rus	1.15805388e-05	0.99998842
rus	3.49035619e-05	0.9999651
rus	5.7168482e-06	0.99999428
rus	1.18034111e-05	0.9999882
rus	1.60860732e-05	0.99998391
eng	0.99979636	0.00020364
eng	0.99979615	0.00020385
eng	0.99972935	0.00027065
eng	0.9997959	0.0002041
eng	0.99979615	0.00020385
rus	5.52566322e-06	0.99999447
rus	1.11128233e-05	0.99998889
rus	1.19122191e-05	0.99998809
rus	5.70699932e-05	0.99994293
rus	1.6217521e-05	0.99998378
eng	0.99979622	0.00020378
eng	0.99979624	0.00020376
eng	0.99979624	0.00020376
eng	0.99978415	0.00021585
eng	0.99838725	0.00161275
rus	2.47732505e-05	0.99997523
rus	0.00030373	0.99969627

rus	0.00013856	0.99986144
rus	8.52489696e-06	0.99999148
rus	4.531384e-05	0.99995469
eng	0.99979634	0.00020366
eng	0.99979628	0.00020372
eng	0.99979456	0.00020544
eng	0.99979628	0.00020372

V. CONCLUSIONS

It can be concluded that out of 25 samples, 23 were correctly recognized, which is 92%. The recognition speed of 1 directory with 10 samples is 0.0269. It has been found that the optimal level of decomposition (decomposition level) is dl = 8 for the performance of the given task. When dl > 8, the quality of learning the neural network significantly deteriorates and drops to 85%. In the case of dl < 8, the performance remains the same as when dl = 8, but the amount of input data increases, causing the recognition speed to decrease.

VI. REFERENCES

- Бондаренко М.Ф. Розпізнавання мови: етапи розвитку, сучасні технології і перспективи їх застосування / М.Ф. Бондаренко, А.В. Работягов, С.В. Щепковський // Біоніка інтелекту: наук.-техн. журнал. – 2010. – № 2 (73). – С. 164–168.
- Винцюк Т. К. Анализ, распознавание и интерпретация речевых сигналов / Т. К. Винцюк. – Киев: Наук. думка, 1987. – 262 с.
- Михайлов В. Ю., Мельников О. Ю. Задача визначення мови вимовленого тексту // Матеріали V Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених за тематикою «Сучасні комп'ютерні системи та мережі в управлінні»: збірка наукових праць / Під редакцією А.А. Григорової. – Херсон: Видавництво ФОП Вишемирський В. С., 2022. – С.148-151. – ISBN 978-617-7941-89-6 (електронне видання)
- Нейронні мережі для початківців. Частина 1. [Електронний ресурс] – URL: https://habr.com/ru/post/312450/ Дата звернення: 25.01.2023.
- Нейронні мережі для початківців. Частина 2. [Електронний ресурс] – URL: https://habr.com/ru/post/313216/ Дата звернення: 25.01.2023.
- Discrete wavelet transform. [Електронний ресурс] – URL: https://en.wikipedia.org/wiki/Discrete_wavelet_transform. Дата звернення: 25.01.2023.
- Мельников О. Ю., Михайлов В. Ю. Застосування нейронних мереж для визначення належності речення до конкретної мови // Нейромережні технології та їх застосування НМТіЗ-2022: збірник наукових праць XXI Міжнародної наукової конференції «Нейромережні технології та їх застосування НМТіЗ-2022» / [за заг. ред. д-ра техн. наук., проф. С. В. Ковалевського і Hon.D.Sc., prof. Dasic Predrag]. – Краматорськ: ДДМА, 2022. – С. 91-93. – ISBN 972-966-379-965-0.