UDC 518.5 **doi:** https://doi.org/10.20998/3083-6298.2025.02.09

Iryna Serdyuk[1], Oleh Tonitsa[1], Oksana Heliarovska[1], Oleksiy Yanovsky[1]

[1] National Technical University “Kharkiv Polytechnic Institute”, Kharkiv, Ukraine

# INFORMATION TECHNOLOGIES OF NEURAL NETWORK SPEECH RECOGNITION IN REAL-TIME

**Abstract. Relevance.** Nowadays, it is relevant to study the basic means of audio signal processing, mainly from the point of view of sound classification and approaches to their improvement. The general characteristics of sound signals are considered, followed by a description of the time-frequency images for sound and the attributes useful for classification are reviewed. Human hearing is an incredible tool that gives us a lot of information about the world around us. We easily catch the sounds of birds, the sounds of cars at a distance, and even complex musical compositions. **The subject of the study** in the article is the human auditory system, which is able to process all this information, analyzing and grouping different sounds. This process is known as auditory scene analysis. Applications such as speech recognition, music transcription, and multimedia data retrieval can be greatly improved by separating and classifying sound sources. Digital audio signal processing has a number of important applications, such as audio compression, sound effect synthesis, and sound classification. Sound classification is becoming increasingly important as more and more multimedia content is created. This is especially useful when it comes to searching through audiovisual materials, as listening to audio clips can be a more efficient way to navigate than watching video scenes. Sound classification can also be used as an interface for audio compression, as different types of sounds, such as music and speech, require different compression methods. **The purpose of this work** is to explore approaches to building neural network speech recognition systems. Real-time speech recognition has become an incredibly useful tool for solving a variety of problems in different areas of life. Many companies now offer dictation software that allows people to create search queries or dictate emails using voice commands. It is appropriate to consider neural network speech recognition, in particular, Ukrainian. One of the biggest problems faced by the analysis of Ukrainian speech is the limited number of models available for recognition. While there are many models for English, there are very few for Ukrainian. In general, the potential benefits of sound processing and speech recognition are obvious, and it is quite likely that we will continue to see new developments in these areas in the future. Neural networks are described, the principle of their operation and methods of audio recognition using them. **The following results were obtained:** the audio signal, its representation, statistical and physical methods of working with it were studied. **Conclusion.** Effective models for correct speech recognition and toolkits for model training were found.

**Keywords:** neural networks, audio signal processing, convolutional neural network, gestalt grouping, cochlear model,dataset

## Introduction

When the air pressure on the eardrum changes, a sound wave is produced, and some of the properties of the sound can be seen in a time-frequency diagram that can help identify its different sources. This definition is based on the Gestalt grouping rules [1], which can also be used to help machines classify and separate sounds. Humans first analyze sound by its frequency, so a time-frequency representation of sound is a useful tool. There are two main ways to visually represent sound: spectrograms and auditory imagery. A spectrogram uses the Fourier transform to analyze the signal, while auditory imagery highlights the most important features based on how humans perceive the sound. Acoustic signal analysis involves using the Fourier transform to create two real-world frequency functions, known as the amplitude spectrum and the phase spectrum. To track changes in the signal over time, Fourier transform spectra of overlapping window segments are calculated at short, successive intervals. However, the phase spectrum is not considered as sensitively as the magnitude or power spectrum. From the effective value spectra, a spectrogram is obtained, which provides a graphical representation of the frequency-time content of the signal.

The auditory representation is usually created by computing an auditory model that captures the physical quantity at a certain point in the auditory pathway. Computational models of hearing simulate the functions of the outer, middle, and inner ears, which work together to transform acoustic energy into neural code in the auditory nerve. These models approximate different stages of auditory processing and are intended to explain the results of psychoacoustic experiments. For example, cochlear models simulate how the basilar membrane filters sound and how this activity is translated into neural activation along the pinna. Various models have been proposed over time, but the cochlear model, which combines sound intensities in different frequency ranges using critical bandpass filters, is popular. These filters correspond to different cochlear channels, which are processed independently of each other at higher levels of the auditory pathway. The cochlear model creates an “auditory image” that is the basis of cognitive functions in the brain. Psychoacoustic observations show that the subjective sensations of spectral components entering the cochlear canal differ from those entering the individual channels

## 1. Mathematical statement of the problem

The human ear can hear sounds with frequencies from 20 Hz to 20 GHz, as long as they do not exceed the "threshold of hearing". The range of sound intensity is enormous (approximately 120 dB): from the noise of rustling leaves to the noise of an airplane taking off. A digital audio signal is obtained by sampling and

quantizing the electrical output of a microphone with a sampling frequency of 44100 Hz, which is commonly used. Sounds can be classified into different categories, such as environmental sounds, artificial sounds, speech and music, etc. Sound events can be described by their temporal and spectral properties. Examples of atomic sound events include short sounds, such as a door slamming, and longer, uniform, textured sounds, such as continuous rain. Temporal properties refer to the duration of the sound and its amplitude modulations, while spectral properties refer to its frequency components and their relative strengths.

Audio signals are complex and diverse, with both spectral and temporal properties affecting their perceptual quality. Representing audio signals requires a joint consideration of both aspects, and short-term analysis is commonly used to estimate signal parameters. Various models can be used to approximate audio signals, such as the source-filter model for speech signals and the sum of elementary components for music. Audio signals are physical stimuli that are processed by the auditory system to evoke psychological sensations in the brain. Perceptual characteristics such as pitch, loudness, subjective duration, and timbre have been studied since Helmholtz in 1870. These sensations are correlated with various spectral and temporal properties of sound, and it is important to consider both when representing audio signals. Short-term analysis is commonly used to estimate signal parameters or features that relate to the underlying signal model. For example, speech signals can be approximated using the source-filter model, while music is modeled as a sum of elementary components.

Audio signals can be periodic or aperiodic. Periodic waveforms are more complex, consisting of a fundamental frequency and a series of overtones, while aperiodic waveforms can contain inharmoniously related sinusoidal tones or a frequency noise waveform. Different amplitudes and phases of the frequency components can affect the overall “color” or timbre of the sound. This is important in speech, where tonal and noise regions alternate according to vowel segments, and in music, where the fundamental frequency and duration can vary greatly. It is useful to study the basic means of audio signal processing, mainly from the perspective of sound classification, to consider the general characteristics of audio signals, followed by a description of time-frequency representations for sound, as well as attributes useful for classification [1,2].

It is necessary to describe the neural networks that are proposed to be used, their principles of operation and methods of audio recognition with their help and to study the audio signal, its representation, statistical and physical methods of working with it. The use of convolutional neural networks, their principles of operation and features are proposed. It is necessary to consider the use of convolutional networks for audio recognition.

## 2. Mathematical model and methods for solving the problem

On the time-frequency diagram, you can see the properties of sound, which can help to identify its different sources. This is based on the Gestalt grouping rules that are used to help classify and separate sounds. People first analyze sound by its frequency, so the time-frequency representation is a useful tool. There are two main ways to visually represent sound: spectrogram and auditory image. The spectrogram uses the Fourier transform to analyze the signal, while the auditory image highlights the most important characteristics based on how people perceive sound. Acoustic signal analysis involves using the Fourier transform to create two real-world frequency functions, known as the amplitude spectrum and the phase spectrum. To track changes in the signal over time, Fourier transform spectra of overlapping window segments are calculated at short successive intervals. However, the phase spectrum is not considered as sensitively as the magnitude or power spectrum. From the rms spectra, a spectrogram is obtained, which provides a graphical representation of the frequency-time content of the signal.

We have considered how to visually represent audio content using spectrograms and audio signals. However, these representations also have many dimensions, which makes them difficult to use for classification. Ideally, we want to extract low-dimensional features from these images or from the audio signal itself that highlight important differences between different types of audios. The international standard for describing audiovisual content, MPEG-7, which is the standard for describing audiovisual content, proposes the use of transformed spectral vectors that are reduced in dimension and decorrelated. A very common method for feature development is to have a complete understanding of the defining features of a signal from both its production and perception perspectives. The goal is to identify features that remain unchanged despite irrelevant changes. Feature extraction is an important aspect of signal processing that involves transforming an audio signal into a numerical representation that describes a particular audio segment. Machine learning algorithms often use input data to partition the feature space into regions, with each region corresponding to a specific class. A comprehensive set of features carefully designed for a specific audio classification task can effectively classify audio signals with a sufficient amount of training data. This is a smaller component of the larger task of auditory scene analysis. When an audio stream consists of many different events from different classes that do not occur simultaneously, splitting the stream into separate events for each class can be achieved by monitoring changes in the values of features typical of the segment boundaries. However, if signals from different sources overlap in time, it becomes much more difficult to separate the streams. Research into sound categorization has led to the development of a vast collection of computational features. These features can be broadly grouped into

two categories: physical and perceptual. Physical features are directly related to measurable properties of an audio signal and are independent of human perception, but perceptual features are subjective and require precise computation using auditory models. These features can be classified as static or dynamic. Static features refer to features of an audio signal that can be captured at a specific point in time by analyzing a short segment of data. In essence, they provide a snapshot of the properties of the signal at a given point in time. Representing static features over a longer period of time leads to better classification. Typically, the time window lasts from 500 milliseconds to 1 seconds. This duration defines the delay to 1 seconds in the task of identifying or categorizing sounds. Physical features refer to signal parameters that capture specific characteristics of an audio signal in terms of time or frequency. Although some of these features may be perceptually determined, they are still considered physical features because they originate from the amplitudes of the audio signal or its short-term spectral values. Let us consider some of the most commonly used physical features. In the equations above, the subscript " " indicates the current window, which is the sample of the data segment $x_r[n]$ of length *N*.

Then, for window analysis, we have:

$$\begin{pmatrix} x_r[n] \\ n=1\ldots N \end{pmatrix} \rightarrow \begin{pmatrix} X_r[k] & f[k] \\ k=1\ldots N \end{pmatrix} \quad (1)$$

The zero-crossing rate (ZCR) is a quantity that determines the number of times a signal crosses the zero axis during a certain time interval, in our case within a certain frame. It is defined as

$$ZCR_r = \frac{1}{2}\sum_{n=1}^{N}\left|sign\left(x_r(n)\right) - sign\left(x_{r-1}(n)\right)\right|, \quad (2)$$

where $sign(x) = \begin{cases} 1, & x \geq 0; \\ -1, & x < 0. \end{cases}$

In the case of narrowband signals such as a sine wave, the ZCR corresponds directly to the fundamental frequency. However, for more complex signals, the ZCR is closely related to the average frequency of the energy concentration. In the case of linguistic signals, the short-term ZCR fluctuates rapidly between the voiced and unvoiced segments due to the different spectral energy concentrations. Conversely, the ZCR of musical signals remains stable over a long period of time.

The short-term energy is the root-mean-square value of the waveform in a given data window and is a representation of the time function that envelopes the signal. It is not only the numerical value that is important, but also the change in the value over time. This option can provide insight into the content and characteristics of the fundamental signal. It looks like

$$E_r = \frac{1}{N}\sum_{n=1}^{N}\left|x_r(n)\right|^2 \quad (3)$$

The energy in a particular frequency range of a signal spectrum can be determined by adding the weighted sum of the power spectrum values in that range:

$$Es_r = \frac{1}{N}\sum_{n=1}^{\frac{N}{2}}\left(X_r[k]W[k]\right)^2, \quad (4)$$

where is $W[k]$ a weighting function with non-zero values in the finite range of indices " ", corresponding to the frequency line. Sharp changes in energy in a musical group indicate a change in tonal composition and help to separate the sound into distinct parts. Typically, these instantaneous changes in energy serve to enhance the projection of the music and emphasize the perceived differences between different segments.

The spectral centroid is the center of gravity of the magnitude spectrum. It serves as a useful metric for analyzing the shape of the spectrum, the center frequency of the spectrum being higher when there is more high-frequency content

$$C_r = \frac{\sum_{k=1}^{\frac{N}{2}} f[k] * \left|X_r[k]\right|}{\sum_{k=1}^{\frac{N}{2}}\left|X_r[k]\right|} \quad (5)$$

Directing the main concentration of signal energy towards higher frequencies results in a brighter sound, and therefore the spectral centroid is closely related to the subjective perception of the brightness of the sound. The fundamental frequency is defined as the periodicity of a waveform in the time domain. It can also be determined by analyzing the spectrum of the signal, which can reveal the frequency of the first harmonic or the spacing between harmonics of a periodic signal. However, when it comes to real musical instruments and human voices, estimating the fundamental frequency is difficult because of the variations in the waveform from one period to the next, and because the fundamental frequency can be weaker than other harmonics. This can cause errors in determining the period, such as doubling or halving the actual value. The autocorrelation function (ACF) of a signal can be used to estimate the periodicity:

$$ACF(\tau) = \frac{1}{N}\sum_{n=0}^{N-1}\left(x_r[n]x_r[n+r]\right). \quad (6)$$

The autocorrelation function (ACF) exhibits peaks at local maxima during its peak period and its multiples. The fundamental frequency of a signal can be determined by taking the reciprocal of the delay "$\tau$" corresponding to the highest peak in a given range. Using shorter time delays instead of longer ones is

beneficial because it helps prevent multiple fundamental frequencies. The harmonic coefficient is represented as a normalized value of the delay over a calculation period and indicates the strength of the periodicity of the signal.

The ability of a person to recognize a sound depends on how the sound is perceived. If there is no existing model for the sound source, perceptual attributes can be used instead to classify and segment it. There are three main perceptual attributes of sound: loudness, pitch, and timbre. Loudness and pitch can be adjusted to make sounds louder or softer, but timbre is a more complex concept that helps distinguish between sounds of the same loudness and pitch. To obtain numerical representations of short-term perceptual parameters, a computational model of the auditory system is used to analyze the shape of the sound. Loudness and pitch, along with their changes over time, are important aspects of perception [3].

Loudness is a measure of how strong a sound signal is perceived to be, and is influenced by various factors, including sound intensity, duration, and spectrum. The physiological basis of perceived loudness is determined by the overall activity of the auditory nerve evoked by sound. Loudness models also take into account the frequency dependence of loudness and how loudness can be additive for different sound components that are separated by their spectrum. Although pitch is a perceptual attribute, it is closely related to the physical characteristic of the fundamental frequency. The way humans perceive pitch is related to the logarithmic representation of the fundamental frequency, which means that a sequential change in pitch in music is actually a sequential ratio of the fundamental frequencies. Most pitch detection algorithms work by extracting the fundamental frequency from an acoustic signal by measuring the periodicity of certain temporal features or detecting the harmonic structure of the spectrum. Just as physical characteristics provide important information for identifying objects, changes in pitch.

Loudness is a measure of the perceived loudness of a sound signal and is influenced by a variety of factors, including sound intensity, duration, and spectrum. The physiological basis of perceived loudness is determined by the overall activity of the auditory nerve evoked by the sound. Loudness models also take into account the frequency dependence of loudness and how loudness can be additive across different sound components that are separated by their spectrum. Although pitch is a perceptual attribute, it is closely related to the physical characteristic of the fundamental frequency. The way humans perceive pitch is related to the logarithmic representation of the fundamental frequency, meaning that a sequential change in pitch in music is actually a sequential ratio of the fundamental frequencies. Most pitch detection algorithms work by extracting the fundamental frequency from an acoustic signal by measuring the periodicity of certain temporal features or by detecting the harmonic structure of the spectrum. Just as physical characteristics provide important information for identifying objects, changes in pitch and loudness over time can also provide clues for recognizing sound sources and determining the consistency of sound over a certain duration. By analyzing the energy levels of certain frequency bands in an audio signal, for example, filtered through gamma filters, we can determine the roughness of the sound and the speed of speech syllables. The multimedia space is largely dominated by speech and music, which are the main areas of interest for humans. To effectively classify sounds, it is important to develop a set of features that correspond to the intended sound categories. These features can be selected based on knowledge of the unique characteristics of the sound, either from a production or perception perspective, or through exhaustive comparative evaluations.

After feature extraction, standard machine learning techniques are used to develop a classifier, such as $k$ nearest neighbor, Gaussian classifier, Gaussian mixture model (GMM) classifier, or neural networks. A significant amount of time is spent collecting and preparing training data, focusing on ensuring that the range of sounds in the training set reflects the size of the sound category. For example, the category of car horns will include a variety of car horns that sound for different durations and in rapid succession. The model extraction algorithm adapts to the size of the data, i.e. a narrower range of examples yields a more specialized classifier.

Let's look at the components of a neural network. The fundamental component of a neural network is a computational unit that can process a set of real numbers as input data and, by performing certain calculations, produce output data. A neural unit is essentially a mathematical function that calculates a weighted sum of the input data, including a bias. Each unit has a unique set of weights and biases that correspond to its inputs. This weighted sum, also known as , can be expressed as the sum of the inputs multiplied by their respective weights, plus the bias

$$z = b + \sum_{i=1}^{N} w_i x_i, \tag{7}$$

where $x_i$ is the set of inputs, $w_i$ is the set of weights, and $b$ is the bias.

We can use a weight vector *w*, a scalar shift *a*, and an input vector to represent *a*. For convenience, we can replace the sum with a dot product:

$$z = w * x + b. \tag{8}$$

In neural networks, instead of using a linear function $x$ as the output signal, neural units apply a $f$ nonlinear function to $z$. This resulting output is called the unit activation value, denoted by $a$. Since we are only considering a single unit, the activation of a node is

essentially the final output of the network, which is called: $y : y = a = f(z)$.

We obtain the sigmoid activation function.

This function has certain advantages: it is differentiable and the output data is in the range $[0,1]$:

$$y = \sigma(z) = \frac{1}{1+e^{-z}}. \quad (9)$$

To obtain the neural unit, we substitute equation (8) into equation (9):

$$y = \sigma(w*x+b) = \frac{1}{1+e^{-(w*x+b)}}. \quad (10)$$

We obtain a diagram that reflects the final representation of the basic neural unit. This particular block takes three input values $x_1, x_2, x_3$ and calculates a weighted sum by multiplying each input value by the corresponding weight $(w_1, w_2, w_3)$. The resulting values are then added to the bias neuron $b$. Finally, this sum is processed using a sigmoid function, which generates a number in the range from 0 to 1. Although the sigmoid function can be used as an activation function, in practice it is not used. Instead, it is recommended to use the function tanh, which is very similar to the sigmoid, but often performs better. It is essentially a variant of the sigmoid function, but has a range from $-1$ to $+1$:

$$y = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}. \quad (11)$$

The rectified linear unit is the most common activation function in machine learning. It is the simplest of the activation functions, taking the value $x$, when $x$ greater than 0, and zero otherwise:

$$y = \max(x,0). \quad (12)$$

Activation functions exhibit a variety of characteristics that make them suitable for various language applications and network architectures. For example, the smooth differentiability of the function tanh and the mapping of unique values to the average make it a desirable option. Conversely, the properties ReLU make it nearly linear. The functions are sigmoid and tanh approach the value 1, when the values $z$ are extremely high, resulting in derivatives close to zero. This creates training difficulties because the error signal gradually becomes too small to be used in training, a problem called the vanishing gradient problem. ReLU do not face this problem, since their derivative for high values is approximately equal to unity and is not very close to zero.

## 3. Consider convolutional neural networks (CNNs)

They are a type of neural network specifically designed to process input images. Despite sharing characteristics with simple neural networks, CNNs have a more specific architecture that consists of two main blocks.

The first block is what distinguishes this type of neural network, as it functions as a feature extractor, performing pattern matching using convolution filtering operations. This initial layer filters the image using several convolution kernels and creates “feature maps” that are then normalized and/or transformed using an activation function. This process can be repeated several times, with each iteration filtering the feature maps obtained from the previous iteration, resulting in the new feature maps being modified and normalized. This process can continue until the values of the last union maps are combined into a vector that defines the output of the first block and serves as the input for the second block.

The second block is located at the end of all neural networks used for classification. The input vector values are subjected to numerous linear combinations and activation functions to obtain a new output vector. This resulting vector contains elements equivalent to the number of classes, with each element representing the probability that the image belongs to a particular class. The values of these probabilities range from to , and their sum is .

The layer parameters are determined by gradient backpropagation, where the cross-entropy is minimized during the training phase. In CNN, these parameters are closely related to the characteristics of the image. A convolutional neural network consists of four different layers, namely the convolutional layer, the pooling layer, the correction ReLU layer, and the full-linking layer.

The convolutional layer is a key element of convolutional neural networks and serves as the initial layer. Its main purpose is to identify a specific set of features in the input images using convolutional filtering. To do this, a window representing the object is moved across the image, and the convolution product is calculated between the object and each part of the scanned image. This feature is then considered a filter, making the two terms interchangeable. Finally, the convolutional layer takes multiple images as input and performs a convolution on each of them using each filter. The pooling layer, which is usually placed between two convolutional layers, serves to create multiple feature maps and merge them together using a reduction process that preserves their important features. This involves dividing the image into regular cells and storing the maximum value in each cell, often using small square cells to retain as much information as possible. This process results in a smaller output size while maintaining the same number of feature maps, which ultimately improves the network’s efficiency and prevents overtraining. The last layer in any neural

network is the full-connection layer, which takes the input vector and creates a new output vector.

This is done by applying a linear combination and possibly an activation function to the input values. The last layer is responsible for classification. It returns a vector that indicates the probability that the input image belongs to a particular class. The input table created by the previous layer represents a feature map for a particular object, with high values indicating the location of the object in the image (with varying degrees of accuracy depending on the association). If the location of the object at a certain point in the image indicates a certain class, then the corresponding value in the table is assigned the appropriate weight. When using CNNs to identify audio samples, the input data must be organized into multiple feature maps, which are then fed into the CNN. This concept is borrowed from image processing applications, where it makes sense to organize the input as a two-dimensional array of pixel values with horizontal and vertical coordinates.

Full weight distribution implies using the same weights at each window position. Convolutional neural networks (CNNs) are often called local because the computations at each window location depend on the features of the neighboring image region. In our context, the input "image" can be thought of as a spectrogram with static, delta, and delta-delta functions, which can be represented as red, green, and blue channels. After creating input feature maps, the convolution and pooling layers apply their respective operations to sequentially create unit activations. The convolution level of units and associations can also be organized into maps, similar to the input layer. In CNN terminology, a pair of consecutive convolution and pooling layers is considered a single CNN layer. Thus, a deep CNN involves a sequence of two or more pairs of layers.

In the convolution layer, several feature maps are associated with each input feature map by a set of local weight matrices $w_{i,j}\,(i=1,\ldots,I;\ j=1,\ldots,J)$. This mapping is achieved by the convolution process, which is a common operation in signal processing. If we assume that all input feature maps $O_i\,(i=1,...,I)$ are one-dimensional, we can calculate the value of each unit in the feature map $Q_j\,(j=1,...,J)$ of the convolution layer by using a special formula:

$$q_{i,m}=\sigma\Big(\sum_{i=1}^{I}\sum_{n=1}^{F}o_{i,n+m-1}w_{i,j,n}+w_{0,i}\Big),\ \ (j=1,\ldots,J) \quad (13)$$

where $o_{i,m}$ is the $m-$unit of the map of the ith input element, is the unit of the ith feature map in the convolution layer, is the element of the weight vector that connects the map of the ith input object to the ith element of the map of the convolution layer. Equation (13) can be written in matrix form:

$$Q_j=\sigma\Big(\sum_{i=1}^{I}O_i * w_{i,j}\Big),\ \ (j=1,\ldots,J), \quad (14)$$

where $O_i$ defines the $i-$th input feature map, $w_{i,j}$ is the local inverted weight matrix to match the definition of the convolution operation. and are vectors if the maps are one-dimensional, $O_i$ and $w_{i,j}$ are matrices if the maps are two-dimensional.

The merging path of the layers is also organized in feature maps, which have the same number of feature maps as its convolutional layers, but each map is smaller. The purpose of the merging layer is to reduce the resolution of the feature map. This means that the units of this layer will act as generalizations of the features of the lower convolutional layer, and since these generalizations will again be spatially localized in frequency, they will also be invariant to small changes in location. This reduction is achieved by applying a merging function to a few cells in a local region of size defined by a parameter called the merging size [4-7]. This is usually a simple function, such as maximization or averaging. The merging function is applied independently to each convolutional feature map. For the maximum merging function, the average merging level is defined as follows, respectively:

$$p_{i,m}=\max_{n=1}q_{i,(m-1)*s+n}, \quad (15)$$

$$p_{i,m}=r\sum\nolimits_{i=1}^{G}q_{i,(m-1)*s+n}, \quad (16)$$

where $G$ is the merging size $s$, and the shift size determines the overlap of adjacent merging windows; $r$ – scaling factor. If the merge windows do not overlap and there are no gaps between them, then maximum merge works better.

Consider the issue of learning weights in convolutional neural networks. The weights in the convolution layer can be learned using the error backpropagation algorithm, but special adjustments are required to account for sparse connections and weight distribution [8-12]. To demonstrate the learning algorithm for CNN layers, we present the convolution operation in the same mathematical form as the full connected ANN layer. This will allow the same learning algorithm to be applied in the same way. If one-dimensional feature maps are used, the convolution operations in equation (14) can be expressed as a basic matrix multiplication by introducing a large sparse weight matrix $W$. This matrix is created by multiplying the basic weight matrix:

$$W=\begin{bmatrix} w_{1,1,1} & \cdots & w_{1,J,1} \\ \vdots & \ddots & \vdots \\ w_{I,1,F} & \cdots & w_{I,J,F} \end{bmatrix}_{I*F*J}, \quad (17)$$

where $W$ consists of $I^{*}F$ rows, where $F$ is the size of the filter, each $I$ row contains $I$ rows for the input

feature maps, and $W$ has $J$ columns representing the weights of the feature $J$ maps in the convolution layer.

The input data maps and the convolution functions are vectorized as vectors $o$ and $q$. A vector row is created from the input function maps $O_i\left(i=1,\ldots,I\right)$:

$$o=\left[v_1\left|v_2\right|\ldots\left|v_M\right.\right], \tag{18}$$

where $v_M$ is the row vector, with the values of the $m-$th frequency range on all function maps , $I$ and $M$ is the number of frequency bands of the input layer. The convolution results calculated in (14) can be expressed as

$$q=\sigma\left(oW\right). \tag{19}$$

Equation (19) has the mathematical form of a full-connection layer, so the weights of the convolution layer can be updated using the backpropagation algorithm. The $W$ looks like:

$$W=\epsilon*o'e. \tag{20}$$

In addition, the error vector can be calculated using the same method or passed to the lower layer using a matrix $W$ containing sparse values. In addition, the bias problem can be solved by including an additional row in the matrix that stores the bias values and reproduces them on all bands of the convolutional layers. In addition, we can also add an element to the vector $o$ with a value of one to handle the bias. The pooling layer does not need to be trained because it has no weights. However, it is necessary to pass error signals to the lower layers through the pooling function. With maximum pooling, for example, the error signal is passed only to the most active block, which is the largest in the group of pooled blocks [13-15]. Therefore, to calculate the error signal that is sent back to the lower convolution layer, the error signal:

$$e_{i,n}^{low}=\sum_m e_{i,m}*\delta\left(u_{i,m}+\left(m-1\right)*s-n\right), \tag{21}$$

where $\delta\left(x\right)$ is the delta function, $1$ has the value , if $x=0$, otherwise zero, and $u_{i,m}$ is the index of the unit with the maximum value among the combined units and is defined as:

$$u_{i,m}=\arg\max_{n=1} q_{i,(m-1)*s+n}\ . \tag{22}$$

## 4. Discussion of results

The research analyzed various aspects of audio signal processing, in particular for the purpose of their further classification and speech recognition. The main attention was paid to the use of neural networks as a promising tool for solving these problems. The study of statistical and physical methods of working with audio signals allowed to form a basic understanding of their characteristics and methods of representation, such as time-frequency images. This became the starting point for the further development of more complex systems based on machine learning. In particular, the principles of functioning of neural networks and methods of their adaptation for audio recognition were investigated, which confirmed their effectiveness in this direction.

Despite the general effectiveness of neural networks, a significant problem was identified regarding their application to the Ukrainian language. The limited number of available models and training sets for the Ukrainian language creates serious obstacles to the creation of high-quality recognition systems. This deficit contrasts with the large number of resources available for the English language, which emphasizes the need for additional research and development in this direction. Overcoming this barrier is key to further implementing Ukrainian language recognition technologies in everyday life, for example, in voice search or dictation systems.

In summary, the results of the work confirm that neural networks are a powerful tool for speech recognition, in particular, for analyzing audio signals. We managed to find effective models and tools for their training, which lays the foundation for future research. However, in order to fully realize the potential of these technologies for the Ukrainian language, it is necessary to expand the base of available models and training data. This opens up a wide field for further scientific developments aimed at creating effective, reliable and accessible Ukrainian language recognition systems.

## 5. Conclusions

The main methods of audio signal processing were studied, mainly from the point of view of sound classification. The general characteristics of sound signals were considered, followed by a description of time-frequency images for sound. Attributes useful for classification were reviewed. Neural networks, the principle of their operation and methods of audio recognition using them were described. The audio signal, its representation, statistical and physical methods of working with it were studied. Effective models for correct recognition of the Ukrainian language and toolkits for training the model were found.

A dataset consisting of more than minutes of audio containing spoken and literary Ukrainian was created, using this dataset a linguistic model was compiled. Software was created that has an interface for ease of use.

To evaluate the result obtained, a comparison was made with existing solutions that allow recognizing the Ukrainian language in real time. It can be noted that the created software can compete with existing solutions, as it has its advantages, but loses in accuracy when there are a large number of words in a sentence for recognition, so there is a need to improve it.

REFERENCES

1. Mikhaylenko, V. M., Tereykovs'ka, L. O., Tereykovs'kyy, I. A. (2017), Neyromerezhevi modeli ta metody rozpiznavannya fonem v golosovomu sygnali v systemi dystantsiynogo navchannya: monografiya, Kyiv, TsP «Komprynt Publ», 120 p.
2. Bondarenko, M. F., Bilous, N. V., Rutkas, A. G. (2004), Komp'yuterna dyskretna matematyka, Kharkiv, «Kompaniya Smit», 480 p.
3. Uosermen, F. (2001), Neyrokomp'yuterna tekhnika: Teoriya I praktyka / Pereklad ukratins'koyu I. Yu. Yurchak, Kharkiv, KhNEU Publ., pp. 88–94.
4. Kryvokhata, A. G., Kudin, O. V., Choporov, S. V. (2000), Neural network mathematical models in problems of sound signal processing, Kyiv, «Helvetica», 120 p.
5. Novotarskyi, M. A., Nesterenko, B. B. (2004), Artificial neural networks: calculations, *Proceedings of the Institute of Mathematics of the National Academy of Sciences of Ukraine*, T. 50, Kyiv, Institute of Mathematics of the National Academy of Sciences of Ukraine, 408 p.
6. Tereykovsky, I. A., Bushuev, D. A., Tereykovskaya, L. O. (2022), Artificial neural networks: basic principles, Kyiv, KPI, 122 p.
7. Korchenko, A., Tereykovsky, I., Karpinsky, N., Tynymbaev, S. (2016), Neural network models, methods and means of assessing the security parameters of Internet-oriented information systems: monograph, Kyiv, «Nash Format», 273 p.
8. Tereykovsky, I. A. (2007), Neural networks in computer information security: monograph, Kyiv, Polygraph Consulting, 209 p.
9. Subbotin, S. O. (2020), Neural networks: theory and practice, Zhytomyr, Publ. O. O. Evenok, 184 p.
10. Rudenko, O. G., Bodiansky, E. V. (2006), Artificial neural networks: Textbook, Kharkiv, LLC «SMIT Company», 404 p.
11. Dmytrienko, V. D., Zakovorotny, O. Yu., Noskov, V. I., Mezentsev, M. V. (2014), Fundamentals of neurocomputing: a teaching and methodological manual for practical classes, Kharkiv, HTMT, 140 p., https://repository.kpi.kharkov.ua/handle/KhPI-Press/45624
12. Tonitsa, O. V., Boeva, A. A., Shynkarenko, D. V. (2021), Using pattern recognition methods in access control systems, *Information technologies: science, engineering, technology, education, health: abstracts of reports of the XXIX International Scientific and Practical Conference MicroCAD-2021*, Part. IV, Kharkiv, NTU «KhPI», pp. 269, https://repository.kpi.kharkov.ua/handle/KhPI-Press/53772
13. Tonitsa, O. V., Reshetnikova, S. M., Gopei, R. V. (2021), Using neural networks in medical diagnostic systems, *Information technologies: science, engineering, technology, education, health: abstracts of the reports of the XXIX International Scientific and Practical Conference MicroCAD-2021*, Part IV, Kharkiv, NTU «KhPI», pp. 271, https://repository.kpi.kharkov.ua/handle/KhPI-Press/80357
14. Tonitsa, O. V., Lotarev, M. S., Reshetnikova, S. M. (2022), Forecasting of emergency situations using neural networks, *Information technologies: science, engineering, technology, education, health: abstracts of reports of the XX International Scientific and Practical Conference MicroCAD-2022*, Kharkiv, NTU «KhPI», pp. 856, https://repository.kpi.kharkov.ua/handle/KhPI-Press/65099
15. Tonitsa, O. V., Popsuyshapka, T. K., Kornil, T. L. (2004), Recognition of visual images based on the results of unmanned aerial vehicle photography, *Information technologies: science, engineering, technology, education, health: abstracts of reports of the International Scientific and Practical Conference MicroCAD-2024*, Kharkiv, NTU «KhPI», pp. 1311, https://repository.kpi.kharkov.ua/handle/KhPI-Press/79508


Received (Надійшла) 28.07.2025
Accepted for publication (Прийнята до друку) 08.08.2025

ВІДОМОСТІ ПРО АВТОРІВ/ ABOUT THE AUTHORS

**Сердюк Ірина Василівна** - доцент кафедри комп'ютерної математики і аналізу даних, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна;
**Iryna Serdiuk -** Associate Professor of the Department of Computer Mathematics and Data Analysis, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine;
e-mail: Iryna.Serdiuk@khpi.edu.ua; ORCID Author ID: https://orcid.org/0009-0001-1143-9145;

**Тоніца Олег Владимирович –** кандидат фізико-математичних наук, доцент, доцент кафедри комп'ютерної математики і аналізу даних, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна;
**Oleh Tonitsa -** Candidate of physical-mathematical Sciences, Associate Professor, Associate Professor of the Department of Computer Mathematics and Data Analysis, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine;
e-mail: Oleh.Tonitsa@khpi.edu.ua; ORCID Author ID: https://orcid.org/0009-0001-8498-0522;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=57289398800.

**Геляровська Оксана Анатоліївна -** доцент кафедри комп'ютерної математики і аналізу даних, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна;
**Oksana Heliarovska -** Associate Professor of the Department of Computer Mathematics and Data Analysis, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine;
e-mail: Oksana.Heliarovska@khpi.edu.ua; ORCID Author ID: https://orcid.org/0000-0002-8927-7465;
**Яновський Олексій Васильович** – аспірант кафедри комп'ютерної математики і аналізу даних, Національний технічний університет «Харківський політехнічний інститут», м. Харків, Україна;
**Oleksii Yanovsky** - graduate student of the Department of Computer Mathematics and Data Analysis, National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine;
e-mail: oleksii.yanovskyi@cs.khpi.edu.ua.

## ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ НЕЙРОМЕРЕЖЕВОГО РОЗПІЗНАВАННЯ МОВЛЕННЯ В РЕЖИМІ РЕАЛЬНОГО ЧАСУ

І. В. Сердюк, О. В. Тоніца, О. А. Геляровська, О. В. Яновський

**Анотація. Актуальність.** В наш час є актуальним дослідження основних засобів обробки аудіосигналу, переважно з точки зору класифікації звуку та підходи до їх удосконалення. Розглянуто загальні характеристики звукових сигналів з наступним описом частотно-часових зображень для звука та переглянуті атрибути, корисні для класифікації. Людський слух – це неймовірний інструмент, який дає нам багато інформації про навколишній світ. Ми легко вловлюємо звуки птахів, звуки машин на відстані та навіть складні музичні композиції. **Предметом дослідження** в статті є слухова система людини, що здатна обробляти всю цю інформацію, аналізуючи та групуючи різні звуки. Цей процес відомий як аналіз слухової сцени. Такі програми, як розпізнавання мовлення, транскрипція музики та пошук мультимедійних даних, можна значно вдосконалити за допомогою розділення та класифікації джерел звуку. Обробка цифрового аудіосигналу має ряд важливих застосувань, таких як стиснення аудіоданих, синтез звукових ефектів і класифікація звуків. В наш час класифікація звуку стає все більш важливою, оскільки створюється все більше і більше мультимедійного вмісту. Це особливо корисно, коли йдеться про пошук серед аудіовізуальних матеріалів, оскільки прослуховування аудіокліпів може бути більш ефективним способом навігації, ніж перегляд відеосцен. Класифікацію звуку також можна використовувати як інтерфейс для стиснення аудіо, оскільки різні типи звуків, такі як музика та мова, потребують різних методів стиснення. **Метою даної роботи** є дослідження підходів до створення систем нейромережевого розпізнавання мовлення. Розпізнавання мовлення в реальному часі стало неймовірно корисним інструментом для вирішення різноманітних проблем у різних сферах життя. Зараз багато компаній пропонують програмне забезпечення для диктування, яке дозволяє людям створювати пошукові запити або диктувати електронні листи за допомогою голосових команд. Доцільним є розгляд нейромережевого розпізнавання мови, зокрема, української. Однією з найбільших проблем, з якими стикається аналіз українського мовлення, є обмежена кількість моделей, доступних для розпізнавання. Якщо для англійської є багато моделей, то для української – їх зовсім мало. Загалом потенційні переваги обробки звуку та розпізнавання мовлення очевидні, і цілком імовірно, що ми продовжуватимемо бачити нові розробки в цих сферах у майбутньому. Описані нейромережі, принцип їх роботи та способи розпізнавання аудіо за допомогою них. **Було отримано такі результати:** досліджено аудіосигнал, його представлення, статистичні та фізичні методи роботи з ним. **Висновок.** Знайдено ефективні моделі для коректного розпізнавання мови та тулкіти для навчання моделі.

**Ключові слова:** нейронні мережі, обробка аудіосигналу, згорткова нейромережа, гештальт-групування, кохлеарна модель, датасет.