A Method of Language Units Classification Oriented to Automated Transcribing

Oksana Kovtun [a], Viacheslav Kovtun [b]

[a] Vinnytsia National Technical University, Khmelnitske Shose str., 95, Vinnytsia, 21000, Ukraine
[b] Vasyl' Stus Donetsk National University, 600-richchya Str., 21, Vinnytsia, 21000, Ukraine

Abstract

Classification of language units (sounds, phonemes, lexemes) is an urgent task of computer linguistics. Its effective solution will allow, for example, to automate of the painstaking and time-consuming work of transcribing speech signals, which is necessary when creating speech corpora. Existing approaches to solving this problem mostly come from the field of automated speech recognition and are characterized by either extremely high requirements for the hardware component of the corresponding information system (with local implementation), or a low level of information security and saturated traffic (with network implementation). We also note that the a priori tendency of such systems to take into account the results of predicting the appearance of language units in speech signals in the process of classifying the first ones becomes a drawback when transcribing speech, for which a sufficiently developed universal background model is not available. In the thesis, a method of classification of language units is proposed, based on the Markov interpretation of parametrized cepstral patterns of the short-term representation of speech signals. The described method formalizes both the computationally efficient process of classifying language units based on the stationary distribution of the hidden Markov model of speech, and the training process of such a model, formulated with an orientation to the rational use of memory. Testing of the proposed method of classifying language units in the balanced metric of qualitative indicators showed its significant advantage over the classical approach in conditions where the number of speakers is relatively small and the size of the training sample is limited compared to the size of the test sample. Also, testing showed that the proposed method outperforms the classical method in terms of time spent on training and classification by at least two orders of magnitude.

Keywords 1

computational linguistics; automated transcribing; classification of language units; Markov chain; language model

1. Introduction

The majority of users and specialists inextricably link the task of converting audio recordings of speech into text with the field of speech recognition [1-5]. At first glance, they are right, but the devil is in the details. For example, the result of the "phonogram-text" transformation of the form "eats shoots and leaves" is semantically different from "eats, shoots and leaves". To notice the difference, the information system must not only use an adequate language model but also distinguish stress and intonation. Another "English case" – "ship" and "sheep" are pronounced the same, but, agree, they mean different things. Therefore, the conversion of "phonogram-text" is a multifaceted scientific problem, the solution of which is still ongoing even for one language. These theses are focused on

IntelITSIS'2023: 4rd International Workshop on Intelligent Information Technologies and Systems of Information Security, March 22–24, 2023, Khmelnytskyi, Ukraine
EMAIL: o.kovtun@donnu.edu.ua (Oksana Kovtun); kovtun_v_v@vntu.edu.ua (Viacheslav Kovtun)
ORCID: 0000-0002-9139-8987 (Viacheslav Kovtun); 0000-0002-7624-7072 (A. 2)

transcribing [3, 6-8]. Automation of this procedure is necessary to solve the ever-present problem of creating representative language corpora.

Many popular speech recognition frameworks can potentially be adapted for automated transcribing. For instance, an excellent framework from the Nvidia corporation is called NeMo [8, 9]. This framework has many useful features in the context of our investigation. For example, the "Punctuation and Capitalization Model" unit determines for each spoken lexeme whether its text equivalent should be written in upper or lower case, as well as which punctuation mark should be used after it. At first glance, NeMo is the answer to all our questions, but it has one big drawback – the entire "smart" component of the framework is implemented on BERT [9-11]. This transformer is not just resource-hungry – even its profile training takes place only on the side of the corporation. Household gaming "supercomputers" are weak for this, not to mention mobile gadgets. The second problem is the closedness of the code and the implementation of most recognition operations "in the cloud", which does not seem rational from the information security point of view.

However, there are analogues with open source [3, 12, 13]: CMU Sphinx, Julius, RWTH ASR, Hidden markov model ToolKit (HTK), and Kaldi. The article [3] gives the results of the comparison of these systems. They were trained on 160 hours of English-language audio recordings and verified on a 10-hour test sample. It was Kaldi who won in accuracy, yielding speed. Kaldi uses Mel-Frequency Cepstral Coefficients [14] and Perceptual Linear Prediction [15] to parameterize short-term phonogram fragments. The Kaldi language model is built based on hidden Markov models [15], a Gaussian mixture model [16] and a deep neural network of the Time-Delay Neural Networks type [17]. The well-known Finite-State Transducer (FST) [3] is responsible for speech-language modelling, and the forward-inverse algorithm [3] is responsible for decoding. As you can see, all these technologies have been known for over forty years. The source of Kaldi's effectiveness is not manufacturability but closed training data. And, again, all these technologies are focused on speech recognition based on transcribed language corpora, and not on automating the process of creating such corpora.

Taking into account the strengths and weaknesses of the mentioned methods, we will formulate the necessary attributes of scientific research.

The ***object*** of investigation is a process of classifying a parameterized pattern of a short-term representation of language units.

The ***subject*** of investigation is the theory of probability and mathematical statistics, the theory of relativity, the theory of pattern recognition, methods of digital signal processing and mathematical programming.

The ***aim*** of the investigation is to formalize the computationally efficient process of classifying language units based on the stationary distribution of the hidden Markov language model.

The ***research tasks*** are:

- to formalize the computationally efficient process of classifying language units based on the stationary distribution of the hidden Markov language model;
- to formalize the process of training the language model, formulated with an orientation towards the rational use of memory, necessary for the classification of language units;
- justify the adequacy of the proposed mathematical apparatus and demonstrate its functionality with an example.

2. Models and methods

2.1. Research Statement

We formalize the process of classifying language units at the junction of the paradigms of digital signal processing [18, 19] and pattern recognition [20, 21]. Let the spoken unitary speech unit $X_t[n]$ (sound, phoneme, lexeme) be stored in the phonogram $X[n]$ for a time interval of duration t [ms]. It is necessary to find the index w of this language unit in the corresponding dictionary of capacity W .

The parameterization of the phonogram $X[n]$ is performed by representing the last one with a sequence of intervals of stationarity with a duration of 10 [ms] each (without overlapping): $i=\overline{1,m_t}$, $m_t=\lfloor t/10 \rfloor$.

The result of parameterization of the contents of the i-th stationarity interval of the original phonogram will be the characteristic vector $s_t^{(i)} \in □$. The procedure for forming a characteristic vector $s_t^{(i)}$ includes a sequence of such operations as:

- spectral alignment of the harmonic component of the analyzed stationarity interval: $\hat{X}_t[n]=X_t[n]-0.95X_t[n-1]$;

- application of the short-time Fourier transform [20] to represent the content of the stationarity interval in the frequency space;

- determination of cepstral characteristics [3, 14, 15, 20] (mean values and standard deviations of p Mel-cepstral coefficients, mean value, standard deviation and slope of the spectral centroid, mean value and standard deviation of the spectral decay) for the analyzed stationarity interval. For visual assessment of the content of the analyzed stationarity interval, we will use the normalized chrominance energy diagram [22].

Parametrization of the phonogram will be preceded by its division into harmonic (tonal) and percussive (transitional) components. The result of the parametrization of the harmonic component of the phonogram $X[n]$ with duration t will be the matrix of characteristic vectors $S_t=\left(s_t^{(1)},\ldots,s_t^{(m_t)}\right)$. Next, vector quantization (k-means method [3, 21]) will be applied to all vectors of the matrix S_t to implement the transition $s_t^{(i)} \rightarrow o_t^{(i)} \in O_t$, where $o_t^{(i)} \in [0,K-1]$ is the corresponding index (word) in the codebook V, which contains K words:

$$V: s_t^{(i)} \rightarrow o_t^{(i)},\ \forall i \in [1,m_t],\ s_t^{(i)} \in □,\ o_t^{(i)} \in □. \tag{1}$$

The focus of our investigation is the definition of a language model λ, which generalizes to a set of models of language units $\lambda^{(j)}$ from a dictionary with capacity W: $\lambda=\left\{\lambda^{(j)}\right\}$, $j=\overline{1,W}$. During training, W discrete hidden Markov models are created:

$$\lambda^{(w)}=\left\{\pi^{(w)},A^{(w)},B^{(w)}\right\},\ w=\overline{1,W}, \tag{2}$$

where $\pi^{(w)}=\left\{\pi_i^{(w)}\right\}$, $i=\overline{1,N^{(w)}}$ is a stochastic vector of initial states, the number of which is limited by the value of $N^{(w)}$; $A^{(w)}=\left\{a_{ij}^{(w)}\right\}$, $i,j=\overline{1,N^{(w)}}$ is a matrix of probabilities of transitions between states; $B^{(w)}=\left\{b_i^{(w)}(k)\right\}$, $i=\overline{1,N^{(w)}}$, $k=\overline{1,K}$ is the matrix of output probabilities.

Based on the defined model λ, the classifier returns the index of the language unit $w_t^* \in W$ for the input parameterized pattern O_t of the empirical phonogram $X[n]$ of duration t. At the same time, the generation probabilities of each of the models of the complex pattern O_t are analyzed. The final decision is made under the rule

$$w_t^*=\arg\max_{w=1,\ldots,W}\left(P\left(O_t \middle| \lambda^{(w)}\right)\right). \tag{3}$$

2.2. Mathematical formalization of the investigated process

Let us investigate analytically the operation of processing the unknown parameterized pattern $O=\left(o^{(1)},o^{(2)},\ldots,o^{(m)}\right)$ by the trained hidden Markov model $\lambda=(\pi,A,B)$. Under such conditions, the model index w and the time of observation of the language unit in the analyzed phonogram t will not be taken into account further.

By definition, the next state of the Markov chain Q_{n+1} is determined only by its current state Q_n. For an a priori non-periodic Markov chain, there is a stationary probability distribution $P=\left\{P_j\right\}$, $j=\overline{1,N}$, where P_j is the probability of the system being in the state j. We maintain the indicator $r_{ij}\left(n\right)$ which represents the probability that at the n-th iteration of the Markov process, the investigated system will be in the state j considering that at the initial moment, the system was in a state i. The parameters P_j and $r_{ij}\left(n\right)$ will be combined by the dependence

$$P_j=\lim_{n\to\infty} r_{ij}\left(n\right). \tag{4}$$

Extending condition (4) to all elements of the set P, we write the system of linear equilibrium equations

$$\begin{cases} \sum\limits_{i=1}^{N} P_i a_{ij} - P_j = 0 \forall j \in \left\{\overline{1,N}\right\}, \\ \sum\limits_{j=1}^{N} P_j = 1, \end{cases} \tag{5}$$

where a_{ij} are elements of the matrix of transition probabilities between states A, described when defining the Markov model (2), $i,j\in\left\{\overline{1,N}\right\}$.

By definition: if the process described by the hidden Markov model is in the state j, then the probability of the value k appearing at the output of the model can be characterized by the stochastic parameter $E\left(k\right)$:

$$E\left(k\right)=\sum_j P_j b_j\left(k\right),\ k\in\left\{\overline{1,K}\right\}, \tag{6}$$

where $b_j\left(k\right)$ are the elements of the matrix of output probabilities B, described when defining the Markov model (2), $j\in\left\{\overline{1,N}\right\}$.

We present the probability of generation by the hidden Markov model λ of the complex parameterized pattern $\dot{O}=\left(\dot{o}^{(1)},\dot{o}^{(2)},\ldots,\dot{o}^{(m)}\right)$ in terms of expressions (4), (5):

$$P\left(\dot{O}\middle|\lambda\right)=\prod_{i=1}^{m} E\left(\dot{o}^{(i)}\right). \tag{7}$$

To increase the computational efficiency of the process of calculating expression (7), we present the latter one in logarithmic form:

$$\ln P\left(\dot{O}\middle|\lambda\right)=\sum_{i=1}^{m}\ln E\left(\dot{o}^{(i)}\right). \tag{8}$$

The characteristic $\ln P\left(\dot{O}\middle|\lambda^{(w)}\right)$ presented by expression (8), calculated for $\forall w\in\left\{\overline{1,W}\right\}$, is generalized by rule (3). We will get:

$$w^*=\arg\max_{w=1,\ldots,W}\left(\ln P\left(\dot{O}\middle|\lambda^{(w)}\right)\right). \tag{9}$$

The classification rule (9) relies on the computational procedure (4)-(6), which, in turn, is based on the trained Markov speech model $\lambda=\left\{\lambda^{(w)}\right\}$, $w=\overline{1,W}$. The component models-elements $\lambda^{(w)}$ are created independently, but in a unified manner: first, the elements of the tuple $\left\langle\pi^{(w)},A^{(w)},B^{(w)}\right\rangle$ are determined, and later (on their basis) the stationary distribution $P^{(w)}$ (expression (5)) and the stochastic vector $E^{(w)}$ (expression (6)) are calculated.

Considering that the Markov model λ $\pi^{(w)}=const\forall\lambda^{(w)}$

$A^{(w)}$ is imposed in terms of the Bakis model [23] so

$\{Q_n = i, Q_{n+1} = i\}$, $\{Q_n = i, Q_{n+1} = i+1\}$, $\{Q_m = i, Q_{m+1} = 1\}$, where m is the final state. The most difficult thing is to choose the values of the matrix of output probabilities $B^{(w)}$. To formalize this process, we will apply the Baum-Welsh algorithm [14, 15]. This iterative algorithm allows finding the local maximum probability $P\left(\dot{O}^{(w)} \middle| \lambda^{(w)}\right)$ for the model $\lambda^{(w)}$ based on the training sample size T_d.

We choose the initiating values of the nonzero elements of the matrix $A^{(w)}$ in such a way as to make the transition $\{Q_n = i, Q_{n+1} = i+1\}$ more likely than $\{Q_n = i, Q_{n+1} = i\}$:

$$\begin{cases} a_{i,i+1} = 1 - a_{i,i} \forall i \in \left\{\overline{1, N-1}\right\}, \\ a_{N,1} = 1 - a_{N,N}, \\ a_{i,i} \le 0.5 \forall i \in \left\{\overline{1, N}\right\}. \end{cases} \tag{10}$$

We choose the initial values of the elements of the matrix $B^{(w)}$ based on the calculated ratio of the value of the parameter $U^{(w)}(k)$, which represents the number of occurrences of the index k in the training sample of the word w, to the total number of indices in this training sample, represented by the value of the parameter $U^{(w)}$:

$$b_i^{(w)}(k) = U^{(w)}(k) / U^{(w)}, \; i = \overline{1, N}. \tag{11}$$

Note that all elements of the matrix $B^{(w)}$ must be greater than zero, which is dictated by the use of a logarithm in expression (9). At the same time, in addition to condition (11), the equality of

$$\sum_{k=1}^{K} b_i^{(w)}(k) = 1 \forall i \in \left\{\overline{1, N^{(w)}}\right\}. \tag{12}$$

must hold for the resulting nonzero initiating elements $B^{(w)}$.

After the initial values of elements of the tuple $\left\langle \pi^{(w)}, A^{(w)}, B^{(w)} \right\rangle$ are determined, the sought parameters $P^{(w)}$ and $E^{(w)}$ are calculated by expression (5) and expression (6), respectively. For further profile use of the model $\lambda^{(w)}$, it is sufficient to store only the calculated values of the vector $E^{(w)}$, which represent the probabilities of generation of all K indices by this model. This circumstance guarantees the rational use of memory by the proposed method of classification of language units.

3. Experiments

Section 2 summarizes the theoretical results in the form of profile information technology for the classification of language units. The input parameters of the technology are the trained language model $\lambda = \left\{\lambda^{(w)} = E^{(w)}\right\}$ $w = \overline{1, W}$

$O = \left(o^{(1)}, o^{(2)}, \ldots, o^{(m)}\right)$

O and each $\lambda^{(w)}$-th model-element of the language model λ

m in the parameterized pattern O. Classification results w^* will be obtained in

$O(mW)$

$O(KW)$, where K is the capacity of the codebook V. Implementation of the proposed technology on the platform of parallel computing has great potential. During the parallel calculation of the characteristic $\ln P\left(\dot{O}\middle|\lambda^{(w)}\right)$ expressed by expression (8) $\forall w \in W$, the classification result w^* will be obtained in $O(m)$ iterations, not in $O\left(mN^{(w)}\right)$ iterations.

Experimental studies of the proposed method of classification of language units were conducted using the materials of two language corpora – the Ukrainian-language corpus GRAC and the English-language corpus TIMIT. The General Regionally Annotated Corpus (GRAC) is a corpus of the Ukrainian language & speech with a volume of more than 800 million lexemes, intended for linguistic research on grammar and vocabulary. GRAC works based on the morphological analysis system developed by the r2u group. This system analyzes the spoken text and determines lemma (lexeme) and tags (grammatical features) for each word form. The distribution of speech material by region of origin (territorial units of Ukraine, Ukrainian-speaking diasporas, etc.) is specific to GRAC. Such an organization allowed us to focus our broadcast material on the Vinnytsia region. This narrowing was chosen deliberately to investigate the functioning of the proposed method in conditions where the number of speakers is relatively small and the size of the training sample is limited compared to the size of the test sample. TIMIT is a speech corpus that contains verified results of phonetic and lexical transcription of spoken American English. Phonetic materials in the TIMIT corpus are grouped by sex, region, and date of recording. A large number of available materials in the TIMIT corpora made it possible to investigate the functioning of the proposed method in conditions where the number of speakers is large, and the volume of the training sample significantly exceeds the volume of the test sample. In Fig. 1, the speech base formed based on the GRAC and TIMIT language corpora are presented in a visual form, which the authors used for experimental investigation of the proposed method of classifying language units.

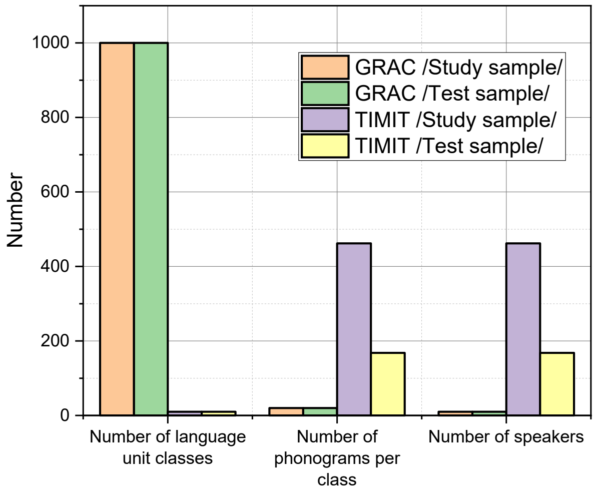

Figure 1: Data structure for empirical research.

The primary metric for evaluating the quality of a classification problem solution is accuracy ρ [18, 19], which is defined as the ratio of the number of cases of correct classification results to the total number of cases of implementation of this procedure. In the terms of the presentation of our investigation, we calculated the metric ρ by the expression

$$\rho = \sum_{t=1}^{T_l} I\left(w^*\left(O_t\right) = y\left(O_t\right)\right) \Big/ T_l \, , \tag{13}$$

where T_l is the total number of attempts to classify language units; O_t is the parameterized pattern of the t-th language unit; $y\left(O_t\right)$ is the true class to which the parameterized pattern O_t of the t-th language unit belongs; $w^*\left(O_t\right)$ is the class to which the author's classification method, generalized by expression (9), assigned the parameterized pattern O_t of the t-th language unit; $I(e)$ is an indicator function: if the argument-condition e is true, then $I(e)=1$ otherwise $I(e)=0$.

We will obtain the empirical functional dependence $\rho = f\left(l, K, N^{(w)}\right)$, where $l = \{GRAC, TIMIT\}$ is the data source, K is the capacity (number of language unit classes) of the code book V, $N^{(w)}$ is the number of states of models $\lambda^{(w)}, w = \overline{1,W}$. The result of the experiment $K = \{2^i\}$, $i = \overline{5,12}$; $N^{(W)} = \{\overline{3,6}\}$ is shown in Fig. 2.

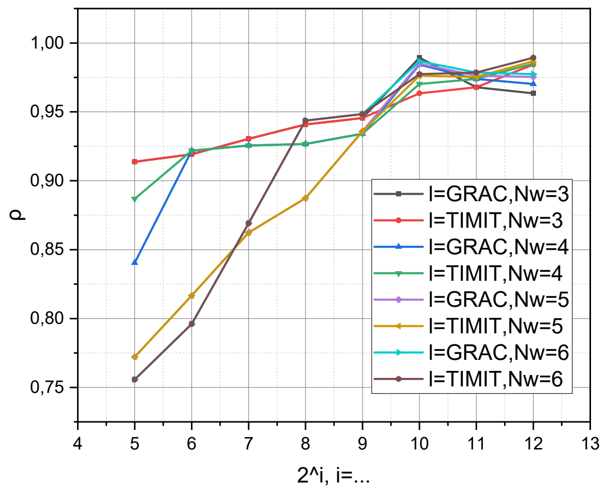

Figure 2: Empirical functional dependence $\rho = f\left(l, K, N^{(w)}\right)$.

Those visualized in Fig. 2 empirical results show that $\max\left(\rho^{GRAC}\right)$ is reached at $\left(K = 2^{10}, N^{(W)} = 3\right)$and $\max\left(\rho^{TIMIT}\right)$ is reached at $\left(K = 2^{12}, N^{(W)} = 6\right)$. Further investigation was conducted with these sets of controlled parameters: $\left(l = GRAC, K = 2^{10}, N^{(W)} = 3\right)$, $\left(l = TIMIT, K = 2^{12}, N^{(W)} = 6\right)$.

In addition to the already defined qualitative characteristic (13), such metrics as precision P, recall R, and F_1-metric are informative for evaluating the quality of the result of solving the classification problem [18, 19]. This "qualitative triple" is calculated based on the content of the confusion matrix $C = \{c_{i,j}\}$, an arbitrary element $c_{i,j}$ which represents the number of cases of classification of the object j as i. Operating with the content of the matrix C, we formulate expressions for calculating $\langle P, R.F_1 \rangle$ in the context of the terminology used in Section 2:

$$P_i = P\left(y(O_t) = i \middle| w^*(O_t) = i\right) = c_{i,i} \Big/ \sum_{j=1}^{W} c_{i,j} \text{ ,} \tag{14}$$

$$R_j = P\left(w^*(O_t) = j \middle| y(O_t) = j\right) = c_{j.j} \Big/ \sum_{i=1}^{W} c_{i,j} \text{ ,} \tag{15}$$

$$F_1^{(i)} = 2P_i R_j \Big/ \left(P_i + R_j\right),\ i, j = \overline{1,W} . \tag{16}$$

Metrics (14)-(16) are defined so that when evaluating an ideal classifier in the qualitative space $\langle P, R, F_1 \rangle$, we obtain $P_i = R_i = F_1^{(i)} = 1$ for $\forall i \in \{\overline{1,W}\}$. Fig. 3 presents an averaged evaluation results of the classical Forward Algorithm method [24, 25] and the author's method in metrics (14)-(16). This experiment took into account the optimal controlled parameters determined by the results of the previous experiment (see Fig. 2).

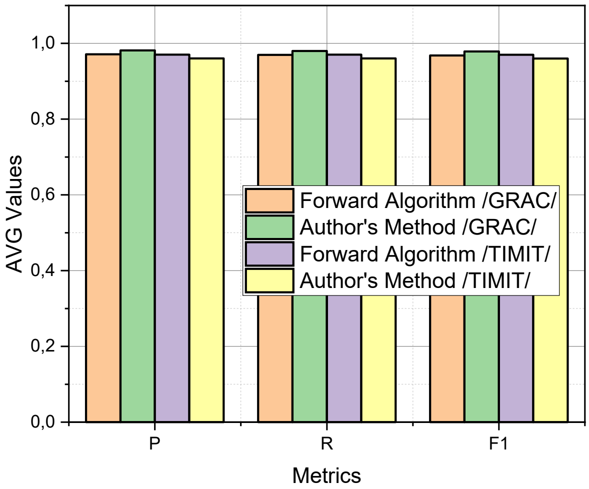

Figure 3: Comparative evaluation of the author's method in metrics (14)-(16)

Note that the total time T spent by the test computer system on the classification of all patterns from $GRAC$ and $TIMIT$ data sets by the Forward Algorithm method FA and the author's method AM amounted to

$$\left\{T_{GRAC,FA},T_{GRAC,AM},T_{TIMIT,FA},T_{TIMIT,AM}\right\}=\left\{0.0845,0.0057,13.9352,0.4521\right\}\text{ [s]}. \tag{17}$$

4. Discussion

Let's start the discussion with the analysis of those shown in Fig. 3 results. These results prove that for the same input data and values of controlled parameters, the author's method and the classical method show close results in the balanced metric of quality parameters (14)-(16). Shown in Fig. 1 information is a confirmation that the amount of input data is statistically representative, and the availability of the source of their origin allows to ensure the reproducibility of the empirical results presented in Section 3. Therefore, the adequacy of the method of classification of language units proposed in Section 2 is empirically proven.

Note that $GRAC$ -data reproduce a situation in which the number of speakers is relatively small, and the size of the training sample is limited compared to the size of the test sample. Instead, $TIMIT$ -data reproduce a situation in which the number of speakers is large, and the volume of the training sample significantly exceeds the volume of the test sample. Fig. 3 shows that in the classification of $GRAC$ -data, the author's method $1.0-1.5\%$ is superior to the classical analogue in all (14)-(16) metrics. At the same time, when classifying $TIMIT$ -data, the classical method prevails. These empirical results can be explained by the fact that when determining the stationary distribution of the hidden Markov model, we deliberately used the most computationally efficient method presented by expression (5). Having gained in computational efficiency, we lost the accuracy of the description of the input data, which begins to manifest itself the more the larger amount of information the trained language model has to generalize (migration from $GRAC$ to $TIMIT$, shown in Fig. 1). However, the author's method is uncompromisingly superior to the classical method in terms of computational efficiency – expression (17) shows an advantage of one and a half to two orders of magnitude in favour of the first one.

The results presented in Fig. 2 are also interesting. It can be seen that the function $\rho=f\left(l=GRAC,K,N^{(w)}\right)$ has an obvious extremum $w=10$. This circumstance can be interpreted in two ways. Either the ratio "number of classes"-"number of instances of the training sample per class" is information lossy, or the k-means method used to implement transition $s_t^{(i)}\to o_t^{(i)}\in O_t$ (see Section 2.1) is suboptimal for generalization. The search for a scientifically reliable answer to this collision is a promising direction for further research.

5. Conclusions

Classification of language units (sounds, phonemes, lexemes) is an urgent task of computer linguistics. Its effective solution will allow, for example, to automate of the painstaking and time-consuming work of transcribing speech signals, which is necessary when creating speech corpora. Existing approaches to solving this problem mostly come from the field of automated speech recognition and are characterized by either extremely high requirements for the hardware component of the corresponding information system (with local implementation), or a low level of information security and saturated traffic (with network implementation). We also note that the a priori tendency of such systems to take into account the results of predicting the appearance of language units in speech signals in the process of classifying the first ones becomes a drawback when transcribing speech, for which a sufficiently developed universal background model is not available.

In the thesis, a method of classification of language units is proposed, based on the Markov interpretation of parametrized cepstral patterns of the short-term representation of speech signals. The described method formalizes both the computationally efficient process of classifying language units based on the stationary distribution of the hidden Markov model of speech, and the training process of such a model, formulated with an orientation to the rational use of memory. Testing of the proposed method of classifying language units in the balanced metric of qualitative indicators showed its significant advantage over the classical approach in conditions where the number of speakers is relatively small and the size of the training sample is limited compared to the size of the test sample. Also, testing showed that the proposed method outperforms the classical method in terms of time spent on training and classification by at least two orders of magnitude.

Further research is planned to be directed at increasing the informativeness of the parameterized pattern of the short-term representation of the phonogram of the speech signal in the context of the task of classifying language units.

6. Acknowledgements

The authors would like to thank the Ukrainian Army for providing security to perform this work. This work has become possible only because of the resilience and courage of the Ukrainian Army.

7. References

[1] P. A. C. Lopes and J. A. B. Gerald, “Iterative MMSE/MAP impulsive noise reduction for OFDM,” Digital Signal Processing, vol. 69. Elsevier BV, pp. 252–258, Oct. 2017. doi: 10.1016/j.dsp.2017.07.002.

[2] V. Kovtun and O. Kovtun, “System of methods of automated cognitive linguistic analysis of speech signals with noise,” Multimedia Tools and Applications, vol. 81, no. 30. Springer Science and Business Media LLC, pp. 43391–43410, May 23, 2022. doi: 10.1007/s11042-022-13249-5.

[3] H. Aldarmaki, A. Ullah, S. Ram, and N. Zaki, “Unsupervised Automatic Speech Recognition: A review,” Speech Communication, vol. 139. Elsevier BV, pp. 76–91, Apr. 2022. doi: 10.1016/j.specom.2022.02.005.

[4] G. Coro, F. V. Massoli, A. Origlia, and F. Cutugno, “Psycho-acoustics inspired automatic speech recognition,” Computers &amp; Electrical Engineering, vol. 93. Elsevier BV, p. 107238, Jul. 2021. doi: 10.1016/j.compeleceng.2021.107238.

[5] M. Malakar and R. B. Keskar, “Progress of machine learning based automatic phoneme recognition and its prospect,” Speech Communication, vol. 135. Elsevier BV, pp. 37–53, Dec. 2021. doi: 10.1016/j.specom.2021.09.006.

[6] B. T. Atmaja, A. Sasou, and M. Akagi, “Survey on bimodal speech emotion recognition from acoustic and linguistic information fusion,” Speech Communication, vol. 140. Elsevier BV, pp. 11–28, May 2022. doi: 10.1016/j.specom.2022.03.002.

[7] M. Diaz-Asper et al., “Using automated syllable counting to detect missing information in speech transcripts from clinical settings,” Psychiatry Research, vol. 315. Elsevier BV, p. 114712, Sep. 2022. doi: 10.1016/j.psychres.2022.114712.

[8] S. Alharbi, M. Hasan, A. J. H. Simons, S. Brumfitt, and P. Green, “Sequence labeling to detect stuttering events in read speech,” Computer Speech &amp; Language, vol. 62. Elsevier BV, p. 101052, Jul. 2020. doi: 10.1016/j.csl.2019.101052.

[9] Z. A. Guven and M. O. Unalir, “Natural language based analysis of SQuAD: An analytical approach for BERT,” Expert Systems with Applications, vol. 195. Elsevier BV, p. 116592, Jun. 2022. doi: 10.1016/j.eswa.2022.116592.

[10] Y. Arase and J. Tsujii, “Transfer fine-tuning of BERT with phrasal paraphrases,” Computer Speech &amp; Language, vol. 66. Elsevier BV, p. 101164, Mar. 2021. doi: 10.1016/j.csl.2020.101164.

[11] J. Zhang, L. Xing, Z. Tan, H. Wang, and K. Wang, “Multi-head attention fusion networks for multi-modal speech emotion recognition,” Computers &amp; Industrial Engineering, vol. 168. Elsevier BV, p. 108078, Jun. 2022. doi: 10.1016/j.cie.2022.108078.

[12] T. Aguiar de Lima and M. Da Costa-Abreu, “A survey on automatic speech recognition systems for Portuguese language and its variations,” Computer Speech &amp; Language, vol. 62. Elsevier BV, p. 101055, Jul. 2020. doi: 10.1016/j.csl.2019.101055.

[13] J. Guglani and A. N. Mishra, “Automatic speech recognition system with pitch dependent features for Punjabi language on KALDI toolkit,” Applied Acoustics, vol. 167. Elsevier BV, p. 107386, Oct. 2020. doi: 10.1016/j.apacoust.2020.107386.

[14] J. Sangeetha, R. Hariprasad, and S. Subhiksha, “Analysis of machine learning algorithms for audio event classification using Mel-frequency cepstral coefficients,” Applied Speech Processing. Elsevier, pp. 175–189, 2021. doi: 10.1016/b978-0-12-823898-1.00009-6.

[15] B. Syiem, S. K. Dutta, J. Binong, and L. J. Singh, “Comparison of Khasi speech representations with different spectral features and hidden Markov states,” Journal of Electronic Science and Technology, vol. 19, no. 2. Elsevier BV, p. 100079, Jun. 2021. doi: 10.1016/j.jnlest.2020.100079.

[16] M. Telmem and Y. Ghanou, “Estimation of the Optimal HMM Parameters for Amazigh Speech Recognition System Using CMU-Sphinx,” Procedia Computer Science, vol. 127. Elsevier BV, pp. 92–101, 2018. doi: 10.1016/j.procs.2018.01.102.

[17] V. V. Kukharchuk et al., “Information Conversion in Measuring Channels with Optoelectronic Sensors,” Sensors, vol. 22, no. 1. MDPI AG, p. 271, Dec. 30, 2021. doi: 10.3390/s22010271.

[18] I. Krak, V. Kuznetsov, S. Kondratiuk, L. Azarova, O. Barmak, and P. Padiuk, “Analysis of Deep Learning Methods in Adaptation to the Small Data Problem Solving,” Lecture Notes in Data Engineering, Computational Intelligence, and Decision Making. Springer International Publishing, pp. 333–352, Sep. 14, 2022. doi: 10.1007/978-3-031-16203-9_20.

[19] I. Dronyuk, O. Fedevych, R. Stolyarchuk, and W. Auzinger, “OMNET++ and Maple software environments for IT Bachelor studies,” Procedia Computer Science, vol. 155. Elsevier BV, pp. 654–659, 2019. doi: 10.1016/j.procs.2019.08.093..

[20] S. Najnin and B. Banerjee, “Speech recognition using cepstral articulatory features,” Speech Communication, vol. 107. Elsevier BV, pp. 26–37, Feb. 2019. doi: 10.1016/j.specom.2019.01.002.

[21] R. Errattahi, A. El Hannani, and H. Ouahmane, “Automatic Speech Recognition Errors Detection and Correction: A Review,” Procedia Computer Science, vol. 128. Elsevier BV, pp. 32–37, 2018. doi: 10.1016/j.procs.2018.03.005.

[22] J. H. Venezia, V. M. Richards, and G. Hickok, “Speech-Driven Spectrotemporal Receptive Fields Beyond the Auditory Cortex,” Hearing Research, vol. 408. Elsevier BV, p. 108307, Sep. 2021. doi: 10.1016/j.heares.2021.108307.

[23] M. Nazarkevych, Y. Voznyi, V. Hrytsyk, I. Klyujnyk, B. Havrysh, and N. Lotoshynska, “Identification of Biometric Images by Machine Learning,” 2021 IEEE 12th International Conference on Electronics and Information Technologies (ELIT). IEEE, May 19, 2021. doi: 10.1109/elit53502.2021.9501064.

[24] P. A. C. Lopes and J. A. B. Gerald, “Iterative MMSE/MAP impulsive noise reduction for OFDM,” Digital Signal Processing, vol. 69. Elsevier BV, pp. 252–258, Oct. 2017. doi: 10.1016/j.dsp.2017.07.002.

[25] Sudhakar and S. Kumar, “MCFT-CNN: Malware classification with fine-tune convolution neural networks using traditional and transfer learning in Internet of Things,” Future Generation Computer Systems, vol. 125. Elsevier BV, pp. 334–351, Dec. 2021. doi: 10.1016/j.future.2021.06.029.