JOURNAL OF MEDICAL INFORMATICS & TECHNOLOGIES Vol. 13/2009, ISSN 1642-6037

automatic speech recognition,
genetic optimization, medical information systems

Jerzy SAS*

OPTIMAL SPOKEN DIALOG CONTROL IN HANDS-FREE MEDICAL INFORMATION SYSTEMS

In the paper a method of optimal selection of utterances used as command entry-words for voice controlled application is presented. Voice controlled programs seem to be particularly useful in the area of medical informatics, where a physician interacts with a program by voice while operating the medical device or being involved in examinations requiring manual activities. The proposed method selects command words from sets of proposals defined for each command so as to minimize the overall probability of incorrect command recognition. First the entry-word dissimilarity matrix is calculated. The word dissimilarities are evaluated using HMM models consisting of appropriately trained acoustic models of the phonemes constituting words. The trained HMM is used as the sample utterance generator for the word. The artificially created utterance samples are then recognized by speech recognizers created for pairs of words. The estimation of correct recognition probability is used as the word dissimilarity measure. The word dissimilarities are then used to determine the average assessment of words selections that can be used as commands. Selection is created by choosing single word from sets of candidates defined for each command. Finally, suboptimal selection is found by using genetic algorithm. Experiments carried out prove that suboptimal selection of command entry-words can observably increase the accuracy of spoken commands recognition in many cases.

1. INTRODUCTION

The optimization of computer software user interface in general consists in minimizing user actions and involvement necessary to control the program. The speech-controlled user interface style seems to be quite natural and convenient. Additionally, it gives the ability to interact with the program without the necessity to engage hands. Such interaction style is especially desirable in case of medical information systems, where the software must be used by a physician while he executes manual activities. Examples of related applications can be found in diagnostic examinations in gastroscopy, proctoscopy, colonoscopy etc. where, while operating the diagnostic imaging devise, the user needs to control some device parameters or enter elements of diagnostic image report into computer. Other similar application areas can be found in dentistry, surgery or pathomorfology.

In order to make speech control usable in the area of medical systems, command recognition accuracy must be extremely high. In typical case the number of commands to be recognized is of the order of several. Although the number of voice commands to be recognized automatically is low, correct recognition of them may be still a problem in the case of noisy environment (as pointed out in [7]) or if speaker independent approach with single common acoustic model must be used.

In order to maximize the accuracy of command recognition, appropriate utterances should be assigned to commands. At the stage of voice command interface design, appropriate words (or short sequences of words) must be selected for individual commands. The word (or word sequence) that invokes the command will be further called *entry-word*. The entry words should be carefully selected in order to be mentally matchable with corresponding command by a human and to be easily distinguishable by the speech recognition engine. In this paper a method of suboptimal selection of entry-words is proposed.

It is assumed that for each command to be recognized the set of entry-word candidates is given. The aim is to make such selection of entry-words, where single entry-word is selected for each command and the selection maximizes the command recognition accuracy. The relevant optimization problem is however difficult in practice due to great number of possible combinations of entry words and due to difficulties in estimating the recognizer error rate for each tested combination. The problem can be simplified by replacing the maximization of recognizer estimated accuracy with maximizing the average pairwise dissimilarity of entry-words. Intuitively, the voice command recognizer is able to accurately recognize the entry-words if they are dissimilar in pairs. Our aim is then to find such entry-words combination than maximizes average words dissimilarity.

The problem of defining words similarity in context of speech recognition was considered in a number of articles. In [6] authors define the word similarity measure based on edit distance of phonetic translations of words. The approach is however not quite suitable for the application being considered here because the acoustic similarities between phonemes are not taken into account. In [8] the similarity of hidden Markov model (HMM) created for compared words is used as the words similarity measure. The authors compare probability distribution functions in corresponding states of HMMs. The approach presented in [2] defines the word similarity measure based on HMM paths co-occurrence. The co-occurrence is the probability that two HMMs being compared follow the same state trajectory. Similar method is proposed in [4], where the word similarity is determined by comparing probabilities of observation sequences generated by two HMMs built for words being compared.

* Institute of Informatics, Wroclaw University of Technology, 50-370 Wroclaw, ul.Wyb. Wyspianskiego 27, email: jerzy.sas@pwr.wroc.pl.

For the sake of entry-word selection problem, where the ultimate aim is to minimize word recognition error rate, the similarity measure that is correlated with the probability of word recognizer accuracy is required. The novel words dissimilarity measure is proposed, which is based on word recognition error rate of the recognizer configured to recognize only two words. Having calculated the matrix of word dissimilarities, the suboptimal selection of entry-words is achieved by applying genetic algorithm.

Before the acoustic signal representing a command is passed to recognition, it must be isolated from continuous stream of data acquired from A/D converter. We do not deal here with the problem of entry-word extraction. Effective methods that can be applied to achieve the appropriate segmentation are described in [9] and [10].

The paper is organized as follows. In the section 2 the common idea of hidden Markov model application to speech recognition and to random creation of artificial utterances is outlined. In the next section the problem of optimal entry-words selection is formulated for two cases: a) where the utterances subject to recognition are merely utterances of command entry-words and b) where the commands are interleaved with natural language utterances as e.g. when using formatting commands applied to dictated text. In the section 4 the method of word similarity assessment is described and details of the optimization procedure based on genetic algorithm are discussed. The experimental results are presented in section 5.

2. APPLIED SPEECH RECOGNITION APPROACH

The method of finding the optimal set of command utterances is based on probabilistic approach used in speech recognizer. Typical approach utilizing acoustic models and language models combined into compound HMM is used ([1, 3, 5]). The model created for the sake of speech recognition is used not only to recognize commands and free speech but also to estimate the probability of utterance misrecognition. Basic notions related to HMM application in automatic speech recognition (ASR) are briefly summarized here.

The ASR procedure uses the probabilistic speech model based on the concept of HMM. HMM is a tuple:

$$HMM = < S, P, B, \pi, S_W > , \tag{1}$$

where S is the set consisting of J states, P is the matrix of state transition probabilities, $B = \{b_1, b_2, ..., b_J\}$ is the set of probability distributions of observation vector emission in states from S, π is the initial distribution of states and $S_W \subseteq S$ is the set of terminal states.

The speech acoustic signal from the sound acquisition device is first segmented into fragments separated by silence. The segments are then processed independently. Each segment is converted into the sequence of observations (o_1, o_2, ..., o_t). The observation is obtained by applying Fourier analysis to the short slice of the input sequence of measured acoustic signal. The speech model conceptually consists of three levels. On the lower level the phonemes are modelled by uniformly structured models consisting of fixed number of states. For each state the corresponding probability distribution of observation vectors is estimated using the training set and by applying Baum-Welsh procedure. The models of words (or fixed sequences of words) are created by concatenating models of subsequent phonemes appearing in the phonetic translation of the word (or the fixed word sequence). Because the phoneme HMMs can be multiply applied in various words, training of HMM for the language consisting of a set of words does not require all words from the language to be presented during training. The last state of each sequence is the terminal state. Let S_W denotes the subset of states $\{s_{w_1}, s_{w_2}, ..., s_{w_N}\}$ being the terminal states of HMMs created for words in the dictionary. Here we consider recognition of utterances coming from fixed set of entry-words. The recognizer is therefore configured to recognize isolated entry-words. In such configuration, word (or sequence) models are linked in parallel as shown on Fig 1.

Typically the model created in the described way is used in speech recognition. The recognition with HMM consists in finding such word w^* which maximizes its conditional probability given the observed sequence:

$$w^* = \arg\max_{w \in D} \; P(w \mid o_1, o_2, ..., o_t) , \tag{2}$$

where D is the set of permissible words. The optimization problem given by (2) is equivalent to finding such state sequence $S^* = (s^*_{i_1}, s^*_{i_2}, ..., s^*_{i_t})$ in compound HMM which ends in state $s_{i_t} \in S_W$ and which maximizes its conditional probability:

$$S^* = \arg\max_{\substack{s_{i_1} s_{i_2}, ..., s_{i_t} \in S^t \\ s_{i_t} \in S_W}} \; P(s_{i_1} s_{i_2}, ..., s_{i_t} \mid o_1, o_2, ..., o_t) . \tag{3}$$

The optimization problem can be transformed by applying Bayes formula to the equivalent one:

$$S^* = \arg\max_{\substack{s_{i_1} s_{i_2}, ..., s_{i_t} \in S^t \\ s_{i_t} \in S_W}} \; P((s_{i_1} s_{i_2}, ..., s_{i_t}) \wedge (o_1, o_2, ..., o_t)) , \tag{4}$$

which in turn can be efficiently solved with Viterbi procedure ([1,3]).

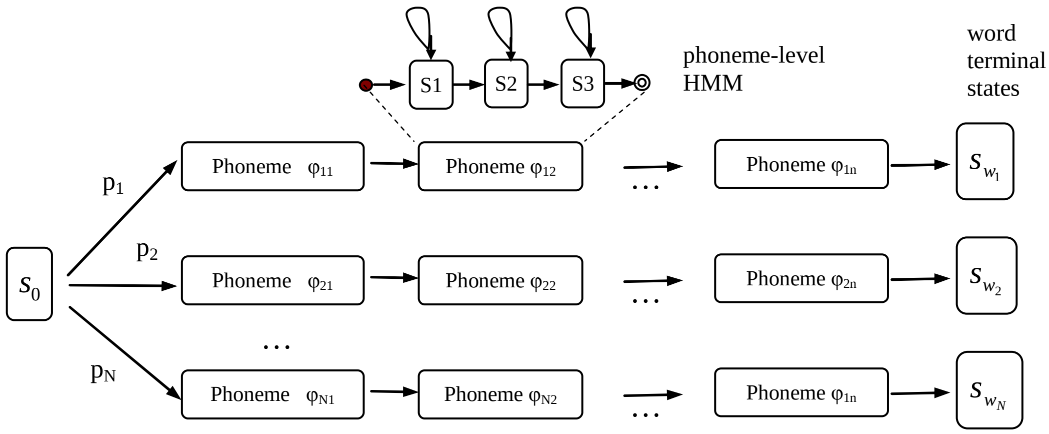

Fig. 1. HMM configured to recognize isolated words or word fixed sequences

The application of described procedure to speech recognition is motivated by the assumption that obtained model is a good approximation of the speaker pronunciation. In other words, if the model is used to create the sequences of observations such that the corresponding HMM trajectory ends-up in one of terminal states in S_W then the distribution of obtained observation sequences is similar the observation sequences extracted from the speech samples of the genuine speaker, for whom the model has been trained. The HMM build for recognition and trained appropriately for a particular speaker (or group of speakers) can be therefore used as the generator of artificial observation sequences similar the ones actually extracted from human speech. This idea can be applied in creation of artificial set of speech samples used to evaluate word similarities, as mentioned in introduction.

3. PROBLEM FORMULATION

3.1. ISOLATED CONTROL COMMANDS RECOGNITION

We will first consider the simple case, where the dialog is restricted to uttering the commands that result in certain operations executed by the program being controlled, like: displaying appropriate dialog windows, setting focus in target edit control, selection of an item from drop-down list, navigating through form elements etc. Each command is invoked by its entry-word. In fact entry-word can be not only single word but also a short fixed phrase consisting of a few words, e.g: "next field", "select disease", "find patient" "close window" etc. For each command only single entry-word can be used.

Let us consider a finite set of commands $C = \{c_1, c_1, ..., c_N\}$. For each command c_n the set of alternate candidates of entry-words is determined:

$$W_n = \{w_1^{(n)}, w_2^{(n)}, ..., w_{I_n}^{(n)}\}\ . \qquad (5)$$

Only single entry-word from each set W_n will be selected to invoke the command. The commands are used with different relative frequencies. Let us assume that the command probability distribution $\pi = (p_1, p_2, ..., p_N)$ is given or can be estimated. The recognizer is provided with correctly segmented input stream, where each segment corresponds to single uttered command. Commands are independent, i.e. current command probability does not depend on the previous command. In result, atomic recognizer action is considered consisting of recognition just single utterance from the set of n acceptable items. The speech recognizer recognizes the commands imperfectly, i.e. no matter what words will be selected to represent the command, it may happen that incorrect command will be recognized. Our aim is to make such selection of entry word for each command in the set C so as to minimize the overall probability of command misrecognition. By a selection S_m we will mean the assignment of words to particular commands:

$$S_m = (w_{i_{1,m}}^{(1)}, w_{i_{2,m}}^{(2)}, ..., w_{i_{N,m}}^{(N)})\ . \qquad (6)$$

The number of possible selections is:

$$M = \prod_{n=1}^{N} I_n\ , \qquad (7)$$

where I_n is the number of candidates in the set W_n. For each selection the probability of command misrecognition can be calculated. Let $\omega_n, n = 1, ..., N$ denote the sequence of feature vectors extracted from the utterance of the command c_n. The recognizer Ψ configured to recognize words from the selection S_m and provided with the sequence ω_n recognizes the command c_j, i.e. $\Psi(\omega_n, S_m) = c_j$. If j=n then the command is recognized correctly, otherwise command recognition error appears. The probability of erroneous recognition for the selection S_m can be calculated as:

$$P_E(S_m) = \sum_{n=1}^{N} p_n[1 - p(\Psi(\omega_n, S_m) = c_n)]. \tag{8}$$

The aim is to find such selection S_{m^*} that minimizes the command erroneous recognition:

$$m^* = \arg\min_{m=1,\dots,M} P_E(S_m). \tag{9}$$

3.2. RECOGNITION OF COMMANDS IN NATURAL LANGUAGE TEXT DICTATION

The problem defined in the previous subsection is relatively simple, because it assumes that the commands subject to recognition are clearly isolated utterances and that the recognition result is always just the command from the defined set *C*. In result, the language model used by the recognizer is extremely simple because the small set of utterances appearing in the applied selection S_{m^*} determines all utterances that need to be considered by the recognition algorithm Ψ. In the case where speech recognition is applied not only to control the application, but also to recognize the speech in natural language, the commands need to be distinguished from the remaining text being dictated. Now we will formulate the more complex problem of command entry-words selection where the commands can be interleaved with the text being dictated. It is assumed that the entry-word will be interpreted as a command merely if it constitutes the complete phrase separated from the rest of the audio input stream by segments of silence. The entry-word constituting the command will be always interpreted as a command.

Although it is neither necessary nor efficient in practical application, for the sake of concept presentation clarity, let's assume that the recognition process consists of two stages. In the first stage it is decided if the observed utterance is a command or it is a natural language phrase from the text being dictated. If the utterance is recognized as a free text, it is subject to second stage recognition where speech recognizer based on full language model is used. We consider here only the first stage of the recognition process.

In order to handle distinguishing between commands and free text, the set of commands *C* is now extended with the new element: "not a command" corresponding to the free language utterance. The extended set $C' = \{c_1, c_1, \dots, c_N, c_X\}$ is further considered. Additionally, the probability p_c that the utterance is a command needs to be estimated by observing real dialogue with the system. New vector of probabilities of commands in the set *C′* can be defined as: $(p'_1, p'_2, \dots, p'_N, p'_{N+1})$, where $p'_n = p_n / p_c$ for $n = 1, \dots N$ and $p'_{N+1} = 1 - p_C$. Finally, the selection problem is analogous to previous one defined by equation (8) and (9) where probabilities p_n are replaced by probabilities p'_n for $n = 1, \dots, N+1$

4. FINDING OPTIMAL COMMAND ENTRY-WORDS SELECTION

The optimal set of entry-words for commands can be found by solving the optimization problem defined in (6). In order to apply it efficiently, the optimization technique and $p(\Psi(\omega_n, S_m) = c_n)$ probabilities estimation method must be proposed.

4.1. ESTIMATING CORRECT COMMAND RECOGNITION PROBABILITIES BY UTTERANCE SIMULATION WITH HMM

Typically to estimate the performance of the recognizer, a testing set is used. By providing the recognizer with samples from the testing set the relative frequency of error can be estimated. In the case of problem being described here, sufficiently numerous set of command utterances is necessary to precisely estimates the probability $P_E(S_m)$. Gathering such testing set can be difficult, in particular when speaker-dependent approach is chosen. Alternate approach can be proposed, which utilizes HMM built for speech recognition. Because as a result of training, HMM becomes the relatively accurate model of a speaker, it can be used as a testing set generator. The observation sequences generated by HMM can be then again recognized by it. In this way virtually unlimited set of artificial utterances can be obtained which may be next used to estimate the command recognition accuracy. Note that in order to estimate recognizer accuracy with the artificial testing set, the acoustic form of the test utterance is not necessary. It is sufficient to collect the observation sequences $(o_1, o_2, \dots, o_t)$, which are the actual input to the recognizer Ψ. Note also that HMMs created for command entry-words do not need to be trained using the recognized commands. Actually, only phoneme HMMs need to be trained, which can be achieved with any utterances, e.g. coming from free language to be recognized. In this way requesting the speaker to provide the testing set specific for commands recognition can be avoided.

The procedure of artificial training set creation can be arranged so as it also randomly draws the command for which the observation sequence will be generated. Let us assume that *N* HMMs have been created for all command entry-words in the selection S_m. In order to create the artificial sample of the entry word for the command c_n, corresponding HMM is selected. The model starts in its initial state. In subsequent steps HMM traverses from state to state until a terminal state is reached. In each step the HMM goes to one of its states and the single observation vector is randomly drawn according to the probability distribution b_i for current state. The drawn observation vector is appended to the observation sequence. The procedure can be formalized as follows:

```
Input: {H1, H2, ..., HN} - the set of N HMMs: arranged to model
                       selected entry-words for commands,
          HR - compound HMM arranged to recognize one
             of selected entry words,
       π - probability distribution of entry-words,
          k - number of artificial utterance samples to  be drawn.
Output:
         kC - number of samples correctly recognized.

kC = 0;
repeat k times
   select randomly one of N commands according to
      commands probability distribution π = (p1, p2, ..., pN);
   let n = index of randomly selected command;
   let O = ();
   let s = initial state of Hn;
   repeat until terminal state of Hn is reached
      draw observation vector according to the distribution bs;
      append drawn observation vector to O;
      draw the next state index according to the transition
         probability distribution defined by n-th row of P;
      let s = drawn state index;
   recognize O with HR
   if result of recognition is cn
      kC = kC + 1;
return kC;
```

The probability $P_E(S_m)$ can be approximated by relative frequency of correct recognition of artificial samples created by the proposed algorithm calculated as k_C/k, where k is the total number of artificial samples drawn randomly and k_C is the number of samples recognized correctly. The proposed procedure can be also used to evaluate the word models dissimilarity by using models pairwise. For two entry-words being considered w_i and w_j the H_R model for only these two words is constructed. The probability distribution π is set to *(1,0)*. In result all observation sequences used in the test will be generated by the model of the word w_i. The dissimilarity q_{ij} of w_i and w_j is calculated as $q_{ij} = k_C/k$.

4.2. FINDING OPTIMAL WORDS SELECTION WITH GENETIC ALGORITHM

The number of possible selections even for relatively small sets of commands and small number of alternate entry-words is so huge. Applying exhaustive search in order to solve the optimization problem (9) is infeasible in practice. Taking into account that the estimation of correct command recognition probabilities $\Psi(\omega_n, S_m) = c_j$ must be carried out individually for each tested selection S_m, the computational cost of exhaustive search is too high to be applicable.

In order to reduce the computational cost of the method, the estimation of command recognition accuracy can be replaced by estimation of the average dissimilarity of entry-words in the selection S_m. The proposed modification can be justified by the observation that if the entry-words in the set being recognized are dissimilar in pairs then the probability of misrecognition is reduced. Let Q be the NxN matrix of entry-words dissimilarities q_{ij} calculated as proposed in section 4.1. Instead of searching for such selection S_{m^*} that maximizes the entry-word recognition error (8), now we search for such selection that maximizes average weighted entry-word dissimilarity calculated in pairs:

$$V(S_m) = \sum_{i=1}^{N} p_i \sum_{\substack{j=1,\dots,N;\\ j\neq i}} q_{ij} \,. \tag{10}$$

The optimization problem (9) is then replaced by the following one:

$$m^* = \arg\max_{m=1,\dots,M} \; V(S_m) \,. \tag{11}$$

The advantage of the modified approach is that the costly computations related to estimating the entry-words recognition accuracy that had to be carried out repeatedly in each step of the optimization procedure applied to solve the original problem (9) can be avoided in the modified procedure. Instead, it is sufficient to prepare the matrix Q before the procedure optimization procedure begins. In each step of the optimization procedure the simple summation formula given in (10) have to be calculated.

In order to avoid exhaustive search of all possible selections, the less expensive suboptimal optimization method is proposed. Genetic algorithm can be applied to find suboptimal solution of (11). The candidate solutions (selections) are represented by chromosomes. The chromosome is a vector of entry-word candidates in the selection as defined in (6). The chromosome thus consists of N genes being the candidate entry-words in the selection. Such chromosome construction is very convenient for crossover operation, because it makes possible to exchange genes independently. The fitness function $F(S_m)$ is directly based on the weighted average word candidates dissimilarity defined in (10):

$$F(S_m) = V(S_m)^{\alpha}. \tag{12}$$

The dissimilarity measure $V(S_m)$ is raised to the power α constant in order to make the parent selection procedure more sensitive to the differences in fitness. The value of α was set experimentally. The fastest convergence of the genetic algorithm to the suboptimal solution was achieved for $\alpha = 15$. The parent selection is based on the roulette wheel principle. The crossover operation consists in selecting genes independently. The gene is randomly selected from two corresponding genes of parents. The gene on n-th position of the chromosome is the entry-word selected to represent n-th command. The probability of gene selection is proportional to its average dissimilarity to entry-words represented by genes of both parents on remaining gene positions. The population consisted of 100 individuals in each generation. The probability of mutation was set to 0.01.

5. EXPERIMENTS

In order to verify the efficiency of the proposed method an experiment has been carried out. The experiment consisted in comparing the recognition accuracy of the spoken commands set in three cases:

a) the set of entry-word candidates was selected randomly from the set of permissible variants,
b) the set of commands was selected using genetic algorithm as described in section 4.2,
c) the entry-words selection was found by random search where as many selections were examined as the number of individuals created during the execution of genetic algorithm.

Results obtained by applying methods a), b) and c) were also compared with the results of applying truly-optimal selection of entry-words candidates. For relatively small command sets and small sets of entry-word candidates the optimal selection was found by exhaustive search.

Speaker-independent approach was taken, i.e. the speech recognizer was trained with speech samples coming from speakers not participating in further experiment. The recognizer was trained with speech samples from 5 speakers. Two experiments were carried out. In the first experiment the total duration of training samples was about 3 minutes per speaker. In the second experiment extended training set was used, where each speaker was requested to provide about 10 minutes of training samples. Other 5 speakers prepared the testing utterances set consisting of recordings of entry-words. Each entry word was recorded by each test speaker 30 times.

The accuracy achieved with methods b) and c) depends on the number of the search space elements tested while searching for the suboptimal solution. In the experiment, the resultant commands recognition accuracy was estimated for various numbers of examined candidates. The results are shown on Fig 2 and 3.

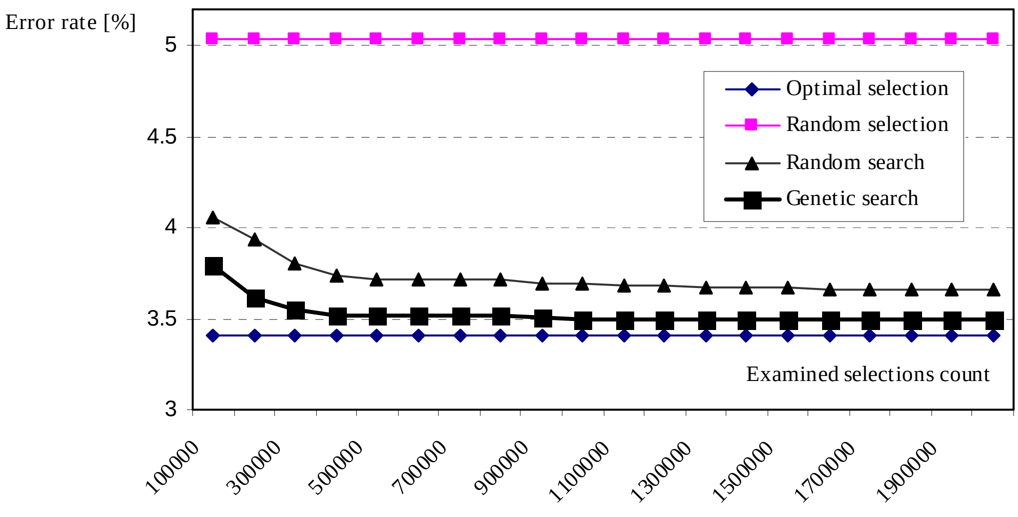

Fig. 2. Comparison of command recognition error rates - small training set (total training time - 15 min)

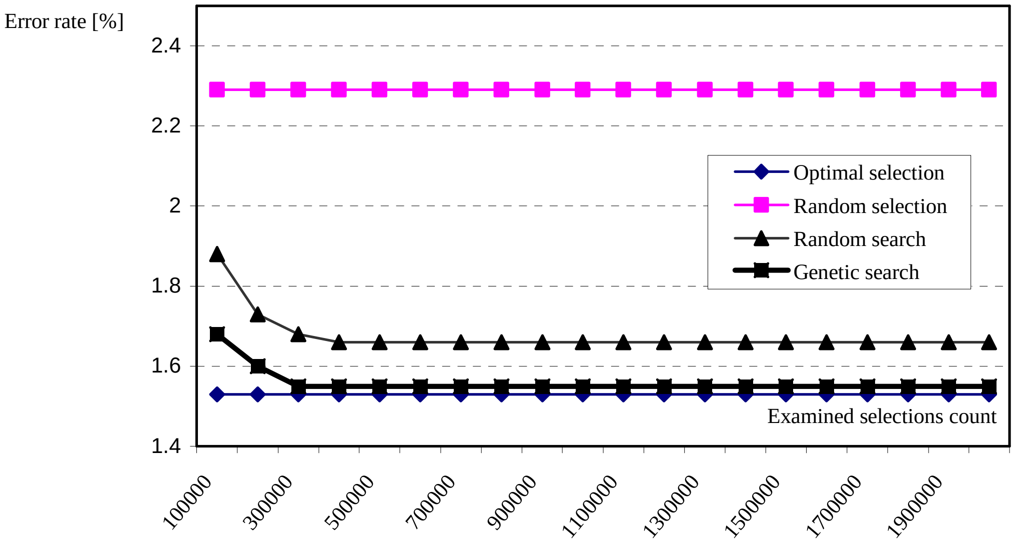

Fig. 3. Comparison of command recognition error rates - large training set (total training time - 50 min)

6. CONCLUSIONS

In the paper the method of optimal spoken command set selection for application in speech-controlled programs is proposed. The method can be applied in medical software used in situations where a physician must control the program or enter information to it having hands free. In applications of this sort the maximal accuracy of the spoken command recognition is crucial.

The experiments carried out proved that by appropriate selection of command entry-words observable reduction of command recognition error rate can be achieved. By applying the method based on genetic algorithm, the relative reduction of error rate at the level of 30% in relation to random selection can be obtained. The error rate close to the one obtained for optimal selection can be obtained very rapidly with genetic algorithm. Acceptable results are achieved just after 1000 iterations (generations) of genetic algorithm. With the proposed method of selection evaluation based on entry-word dissimilarity execution of 1000 iterations of genetic algorithm takes only a few second on modern computer, thus acceptable selection can be obtained almost immediately. Comparison of the results obtained for less and more accurate acoustic model (presented on Fig 2. and Fig 3) indicates that the relative reduction of error rate does not significantly depend on the quality of the acoustic model.

The application of the entry-words selection method proposed here is not restricted to control commands. Similar problem appears in case of spelling out-of-vocabulary symbols being a mixture of letters and digits. The accuracy of the symbol recognition depends strongly on selection of words representing individual letters (usually popular human names are used). The proposed method can be applied to selecting optimal set of names representing letters for the sake of spelling.

BIBLIOGRAPHY

[1] JELINEK F., Statistical Methods for Speech Recognition, MIT Press, Cambridge, Massachusetts, 1997.

[2] LYNGSO, R.B. PEDERSEN C., NIELSEN H., Metrics and Similarity Measures for Hidden Markov Models, Proc. of the 7th Int. Conf. on Intelligent Systems for Molecular Biology (ISMB), pp. 178-186, AAAI Press USA, 1999.

[3] JURAFSKY D., MARTIN J., Speech and Language Processing. An Introduction tp Natural Language Processing, Computational Linguistics and Speech Recognition, Prentice Hall, New Jersey, 2000.

[4] LI, W.; KUBICHEK, R.F., Output-based Objective Speech Quality Measurement Using Continuous Hidden Markov Models, Signal Processing and Its Applications, 2003. Proc. of Seventh Int. Symposium on, Signal Processing and Its Applications, pp. 389 - 392, 2003.

[5] YOUNG S., EVERMAN G., The HTK Book, Cambridge University Engineering Department, 2005

[6] MACKAY W., KONDRAK G., Computing Word Similarity and Identifying Cognates, with Pair Hidden Markov Models Proc. of the 9th Conference on Computational Natural Language Learning (CoNLL), pp. 40–47, 2005.

[7] KACALAK W, MAJEWSKI W., Automatic Recognition of Voice Commands in Natural Language Given by the Operator of the Technological Device Using Artificial Neural Network, In: Kurzynski M., Puchala E., Wozniak M., Zolnierek A., Proc. of 4th Int. Conf. on Computer Recognition Systems, CORES, 05, pp. 689-696, Springer Verlag, 2005.

[8] MOHANTY B., HERSHEY J., OLSEN P., KOZAT S., GOEL V., Optimizing Speech Recognition Grammars Using a Measure of Similarity Between Hidden Markov Models, Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 4953 - 4956, 2008.

[9] PORWIK P., Isolated Word Descriptors and Control Parameters of the Computer Applications. Journal of Medical Informatics & technologies, Vol 10, pp. 35-46, 2006

[10] PORWIK P., PROKSA R., Word Extraction Method in Human Speech Processing, Journal of Medical Informatics & technologies, Vol 12, pp. 209-216, 2008