Research Article

Yinchun Chen*

A hidden Markov optimization model for processing and recognition of English speech feature signals

https://doi.org/10.1515/jisys-2022-0057
received January 19, 2022; accepted April 22, 2022

Abstract: Speech recognition plays an important role in human–computer interaction. The higher the accuracy and efficiency of speech recognition are, the larger the improvement of human–computer interaction performance. This article briefly introduced the hidden Markov model (HMM)-based English speech recognition algorithm and combined it with a back-propagation neural network (BPNN) to further improve the recognition accuracy and reduce the recognition time of English speech. Then, the BPNN-combined HMM algorithm was simulated and compared with the HMM algorithm and the BPNN algorithm. The results showed that increasing the number of test samples increased the word error rate and recognition time of the three speech recognition algorithms, among which the word error rate and recognition time of the BPNN-combined HMM algorithm were the lowest. In conclusion, the BPNN-combined HMM can effectively recognize English speeches, which provides a valid reference for intelligent recognition of English speeches by computers.

Keywords: speech recognition, hidden Markov model, back-propagation, Mel-frequency cepstral coefficient

1 Introduction

With the development of computer technology and virtual reality, people's lives have become increasingly convenient, but the emergence of computer-related industries also means that human–computer interaction has become more frequent [1]. Human–computer interaction can take various forms, including directly inputting commands of texts, images, and facial expressions, among which the easiest and most direct way is voice commands. When using speech for human–computer interaction, the computer needs to "understand" the speech and convert it to the corresponding text through speech recognition technology [2]. However, for speech consisting of plural words, the recognition difficulty is greatly increased because the number of speech features increases on the one hand. On the other hand, the pronunciation habits of the speech signal providers vary when speaking long speeches, so there will be differences at different times and in different environments even if the same person speaks the same speech [3]. These differences do not affect people, but for computers, they can interfere with speech recognition. Therefore, a large number of training samples are needed to build acoustic models for speech recognition. Kim and Stern [4] proposed a feature extraction algorithm and found through experiments that the speech recognition accuracy under this method was significantly improved. Watanabe et al. [5] proposed an automatic speech recognition algorithm and verified the effectiveness of the method through experiments. Sun et al. [6] introduced an unsupervised deep domain adaptive (DDA) acoustic modeling method. The method learned two

* **Corresponding author: Yinchun Chen,** College of Foreign Languages, Shanghai Jian Qiao University, Shanghai 201306, China, e-mail: naochenmi273842@126.com

discriminative classifiers using a deep neural network (DNN), and speech recognition experiments on noise/channel distortion and domain shift verified the effectiveness of the method. Bhatt et al. [7] put forward to improve the monophone-based Hindi connected word speech recognition model with a hidden Markov model (HMM) and found that the improved recognition model gave better results. Yavuz and Topuz [8] designed and implemented a speaker-related, phoneme-based Turkish word recognition system and experimentally verified the recognition accuracy of the system for Turkish words. Lee et al. [9] proposed an improved HMM-based adaptive method for low-frame rate speech recognition and found through experiments that the method could obtain better recognition accuracy. Veisi and Mani [10] combined the deep brief network used for extracting speech signal features and the deep bidirectional long short-term memory with a connectionist temporal classification output layer. They found that the deep neural network could improve recognition performance. Kuanyshbay et al. [11] suggested a method that takes a pretrained model of the Russian language and applies its knowledge as a starting point to a new neural network structure for the automatic recognition of Kazakh speech. The results showed that the pretrained neural network had better recognition performance than the non-pretrained one. This article briefly introduced the HMM-based English speech recognition algorithm and combined it with a back-propagation neural network (BPNN) to further improve the recognition accuracy and reduce the recognition time for English speeches. Then, a simulation experiment was carried out on the BPNN-combined HMM algorithm, and it was compared with HMM and BPNN algorithms.

2 English speech recognition algorithm

2.1 HMM-based English speech recognition

When speech recognition technology is applied, it is generally divided into two stages: the training stage and the recognition stage. There are various types of speech recognition algorithms, but the process of recognizing speech is preprocessing the original speech signal, extracting signal features, training with features (matching calculation), and recognizing results.

The training process of the HMM [12] used in this article is shown in Figure 1.

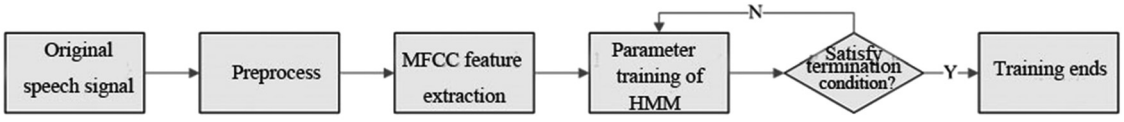

Figure 1: The training process of HMM.

① The original speech signal is input and preprocessed. The purpose of preprocessing is to convert the original continuous speech analog signal into a discrete speech digital signal [13] and remove some of the interference in the original signal. The speech analog signal is converted into a digital signal using sampling and quantization, and the digital signal is windowed using the Hamming window [14]. The corresponding equation is expressed as follows:

$$\begin{cases} S_w(n) = s(n) \times w(n), \\ w(n) = \begin{cases} 0, & \text{else}, \\ 0.54 - 0.46\cos\left(\dfrac{2\pi n}{N-1}\right), & 0 \le n \le N-1, \end{cases} \end{cases} \tag{1}$$

where $S_w(n)$ is the speech digital signal after windowing, $s(n)$ is the original speech digital signal, $w(n)$ is the window function, and N is the length of the digital signal.

② Features are extracted from the preprocessed speech signal. This article adopted the Mel-frequency cepstral coefficient (MFCC) [15] to extract the signal features. The related equations are as follows:

$$\begin{cases} Y(k) = \sum_{n=0}^{N-1} y(n) \cdot e^{\frac{-2j\pi kn}{N}} \\ P(\omega) = |Y(k)|^2 \\ S(m) = \ln\left(\sum_{k=0}^{N-1} P(\omega) \cdot H_m(k)\right) \\ H_m(k) = \begin{cases} 0, & k < f(m-1) \\ \dfrac{2(k - f(m-1))}{(f(m+1) - f(m-1))(f(m) - f(m-1))}, & f(m-1) \le k \le f(m) \\ \dfrac{2(f(m+1) - k)}{(f(m+1) - f(m-1))(f(m) - f(m-1))}, & f(m) < k \le f(m+1) \\ 0, & k > f(m+1) \end{cases} \\ \sum_{m=0}^{M-1} H_m(k) = 1 \\ c(l) = \sum_{m=1}^{M-1} S(m) \cos\left(\dfrac{\pi l(2m+1)}{2M}\right), \quad l = 1, 2, 3, \cdots, L, \end{cases} \tag{2}$$

where $Y(k)$ is the frequency domain signal after a fast Fourier transform, $y(n)$ is the time domain signal of the Mel frequency of English speeches, k is the serial number of the sampling point of the time domain signal when converting the time domain signal to the frequency domain signal, up to N, n is the time sampling point of the time domain signal, $P(\omega)$ is the instantaneous energy of $Y(k)$, $H_m(k)$ is the frequency response of the triangular filter, m is the serial number of a triangular filter in a set, up to M, $f(m)$ is the center frequency of the mth triangular filter in a set of triangular filters, $c(l)$ is the L-order MFCC feature parameter, and $S(m)$ is the energy spectrum function of the frequency domain signal after the filtering process.

③ The HMM is trained using the MFCC features of the digital signal, and the mathematical model of the HMM is expressed as follows:

$$\begin{cases} M = \{S, O, A, B, \pi\} \\ A = \{a_{ij}\} \\ a_{ij} = P(s_t = j | s_{t-1} = i) \ge 0 \\ \sum_{j=1}^{N} a_{ij} = 1 \\ B = \{b_{ij}\} \\ b_{ij}(o) \ge 0 \\ \sum_{o} b_{ij}(o) = 1 \\ \pi = \{\pi_i\} \\ \pi_i = P(s_1 = i) \ge 0 \\ \sum_{j=1}^{N} \pi_i = 1, \end{cases} \tag{3}$$

where M is the HMM, S is the set of hidden states of the HMM, O is the set of observation vectors, A is the set of jump probabilities of the hidden-state self-hop and next-hop, B is the probability density of the observed values, π is the probability value of the state initialization of the HMM, a_{ij} is the transition probability of

transforming from state i (hidden state s_{t-1} at time $t-1$) to state j (hidden state s_t at time t), and $b_{ij}(o)$ is the state probability of the output [16].

④ It is seen from the mathematical model of the HMM, for a given observation sequence O, that the hidden-state S is also fixed, and the values of A, B, and π are unique, i.e., a_{ij}, $b_{ij}(o)$, and π determine the HMM. The iterative adjustment is performed on a_{ij}, $b_{ij}(o)$, and π to make the output probability $P(O|M)$ of O maximum. The formula is expressed as follows:

$$\begin{cases} \xi_i(i,j) = \dfrac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j}^{N}\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)} \\ \hat{a}_{ij} = \dfrac{\sum_{t=1}^{T-1}\xi_i(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)} \\ \hat{b}_{jk} = \dfrac{\sum_{t=1,o_t=v_k}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)} \\ \hat{\pi}_t = \gamma_t(i) \\ \gamma_t(i) = \dfrac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N}\alpha_t(j)\beta_t(j)}, \end{cases} \tag{4}$$

where $\hat{a}_{ij}$, $\hat{b}_{jk}$, and $\hat{\pi}_i$ are the parameters after one iteration, $\alpha_t(i)$ and $\beta_t(i)$ are the forward and backward probabilities, $\xi_t(i,j)$ is the probability of being in the state i at time t and being in the state j at time $t+1$, and $\gamma_t(i)$ is the probability of being in the state i at time t.

⑤ After iteration, whether the HMM satisfies the condition of terminating iteration is determined. If it does, the iteration stops, the parameters of the HMM are output, and the model and the corresponding speech feature data are stored in the speech reference template library; if it does not, it returns to step ④ for iteration.

2.2 The BPNN combined HMM algorithm for recognition

The basic principle of the HMM for speech recognition is based on the statistics of training samples, i.e., the model estimates the hidden state through directly observed sample data; therefore, the established model will not fit the actual states completely and may produce recognition errors in the practical application. In short, the HMM has a strong modeling ability for speech samples but is weak to classify data using modeling, especially when facing a large number of samples [17].

In addition to using the HMM for speech recognition, deep learning neural networks are also used for speech recognition. Deep learning neural networks have the advantages of strong adaptive ability and stronger prediction accuracy with the help of the laws of deep mining. In addition, they can process a large amount of sample data in parallel, i.e., it has more advantageous for big data classification.

The advantage of neural networks in classifying big data can make up for the weakness of HMM in classifying data. First, English speech samples were modeled using an HMM. Then, the English samples were decoded by Viterbi decoding using the established model to obtain the cumulative output probability of the state changes of English speech samples. The cumulative output probability was taken as the input to train a BPNN. The specific process is shown in Figure 2.

① The original English speech signals were postpreprocessed for MFCC feature extraction.

② The parameters in the HMM were iteratively adjusted using the MFCC features of the English speech signals until the termination condition was satisfied.

The aforementioned two steps are HMM modeling for English speech samples. The specific steps and the formulas used are consistent with the training steps of the HMM in Section 2.1, and hence, they are only briefly described here.

③ After model training, the English speech signals were decoded by Viterbi decoding to obtain the cumulative output probability of state changes of the English speech signals.

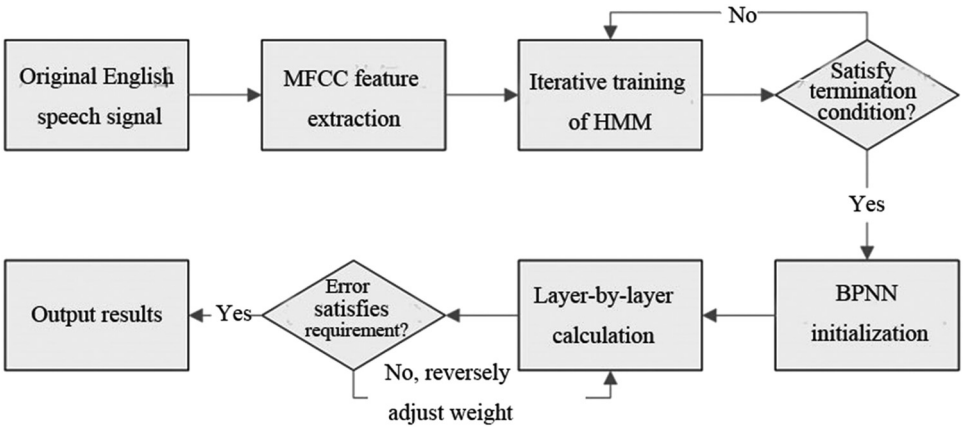

Figure 2: Basic flow of the BPNN-improved HMM algorithm for English speech recognition.

④ The parameters of the BPNN were initialized, and the cumulative output probability of state changes of the English speech signals were input.

⑤ The cumulative probability was calculated layer by layer in the hidden layer of the BPNN by the following formula:

$$a = f\left(\sum_{i=1}^{n} \omega x_i - \beta\right), \tag{5}$$

where x_i is the cumulative output probability of state changes of the input English speech samples, a is the output of every layer, β is the adjustment term of every layer, $f(\cdot)$ is the activation function [18], and ω is the weight between layers.

⑥ The result obtained after layer-by-layer calculation was compared with the actual result of the English speech samples, and the cross-entropy was used as the error between the two results. Whether the error met the requirement, i.e., it converged to stability or iterations reached the set number, was determined. If it did not, the weight parameter in the hidden layer was adjusted reversely, and step ⑤ was repeated; if it did, the training stopped, and the results were output.

The basic process of using the trained BPNN-combined HMM algorithm for English recognition is similar to the training process, but the repeated adjustment of internal parameters is not needed. First, the cumulative output probability is obtained by decoding the English speech samples using Viterbi decoding. Then, the cumulative probability is calculated layer by layer in the trained BPNN to obtain the recognition result.

3 Simulation and analysis

3.1 Experimental environment

The test configurations were Windows 7 system, I7 processor, and 16 G memory.

3.2 Experimental data

English speech data were crawled from the Internet [19], and 10,000 speeches with clean and standard pronunciations were selected, of which 90% were used as the training set and 10% as the test set. The sampling rate was set as 16 kHz, and 16-bit encoding was used. Some sentences were as follows:

① It's a nice day today;
② How can I get to the airport, please;
③ Wha's the price of this product
...

3.3 Experimental setup

The design of structural parameters for the HMM is as follows: the number of states of the speech samples was set as 6 by orthogonal comparison experiments, and the number of possible observations for every state was 4. The initial state distribution transition matrix is given as follows:

$$A = \begin{bmatrix} 0.3 & 0.7 & 0 & 0 & 0 & 0 \\ 0 & 0.3 & 0.7 & 0 & 0 & 0 \\ 0 & 0 & 0.3 & 0.7 & 0 & 0 \\ 0 & 0 & 0 & 0.3 & 0.7 & 0 \\ 0 & 0 & 0 & 0 & 0.3 & 0.7 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6}$$

The relevant parameters of the BPNN are set as follows according to the data volume of single detection and orthogonal experiments. The number of nodes in the input layer was set as 1,024, the number of nodes in the output layer was set as 512, the number of nodes in the hidden layer was set as 2048, the learning rate was set as 0.2, and the error threshold was set as 10^{-7}.

To verify the performance of the BPNN-improved HMM algorithm, it was compared with the HMM algorithm and the BPNN algorithm.

The recognition performance of the three speech recognition algorithms was compared under the different number of test samples. In addition, Gaussian white noise was added to the test set containing 1000 samples to compare the performance of the speech recognition algorithms for test samples with different signal-to-noise ratios and to test the anti-interference abilities of the three recognition algorithms.

3.4 Evaluation indicators

The results obtained after recognition by the speech recognition algorithms were evaluated by the word error rate [20], and the calculation formula is:

$$\mathrm{WER} = \frac{X + Y + Z}{P} * 100\%, \tag{7}$$

where X is the number of substituted words, Y is the number of deleted words, Z is the number of inserted words, and P is the total number of words.

3.5 Experimental results

Figure 3 shows the word error rates of the three speech recognition algorithms under different numbers of test samples. Figure 3 shows that the word error rate of all three speech recognition algorithms tended to decrease and eventually tended to be gentle as the number of test samples increased.

Figure 3 shows that the word error rate of the BPNN-improved HMM algorithm for English speech recognition was always lower than that of the other two recognition algorithms; the word error rate of the BPNN algorithm was higher than that of the HMM when the number of test samples was less than 400, and

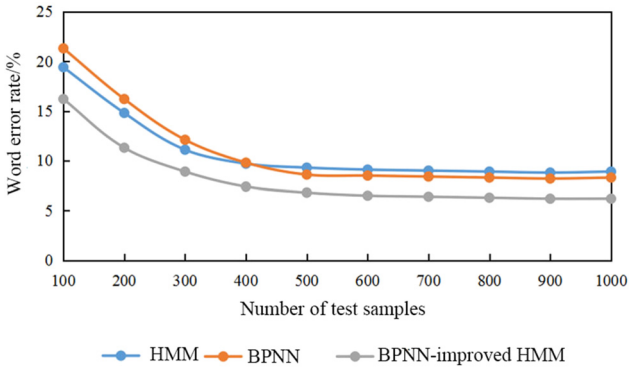

Figure 3: Word error rates of three speech recognition algorithms under different number of test samples.

the word error rate of the HMM algorithm was higher than the BPNN algorithm when the number of test samples was more than 400.

Figure 4 shows the time consumption of the three speech recognition algorithms for speech recognition under the different number of test samples. Figure 4 shows that as the number of test samples increased, the time required by all three recognition algorithms increased. When the number of test samples was less than 400, the recognition time of the three algorithms was relatively close; when it exceeded 400, the gap between the recognition time of the three algorithms became larger, and the increased amplitude also increased. Overall, the HMM algorithm had the longest recognition time, the BPNN had the second-longest recognition time, and the BPNN-improved HMM algorithm had the lowest recognition time under the same number of test samples.

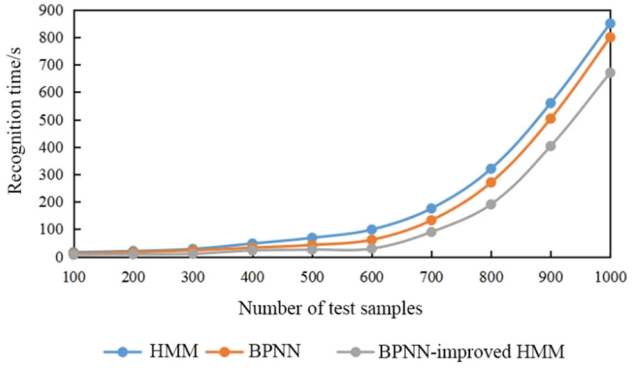

Figure 4: Recognition time consumption of three speech recognition algorithms under different number of test samples.

Figure 5 shows that the word error rate of all three recognition algorithms tended to decrease as the signal-to-noise ratio of the speech signal increased, which meant that the smaller the noise interference was, the lower the word error rate was. Under the same signal-to-noise ratio, the HMM algorithm had the highest word error rate for speech recognition, followed by the BPNN algorithm and the BPNN-improved HMM algorithm.

Figure 6 shows that the recognition time of the three speech recognition algorithms reduced as the signal-to-noise ratio increased. The reason for the aforementioned result is that when the signal-to-noise ratio was higher, there was less interference in the speech signal, and the algorithm was less hindered in recognition. Under the same signal-to-noise ratio, the HMM algorithm had the longest recognition time, followed by the BPNN algorithm and the BPNN-improved HMM algorithm.

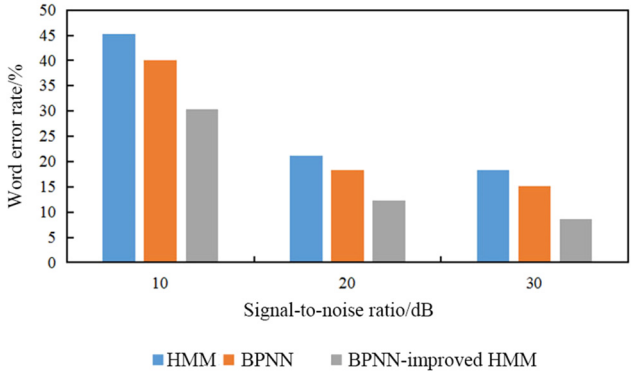

Figure 5: Word error rates of three speech recognition algorithms under different signal-to-noise ratios.

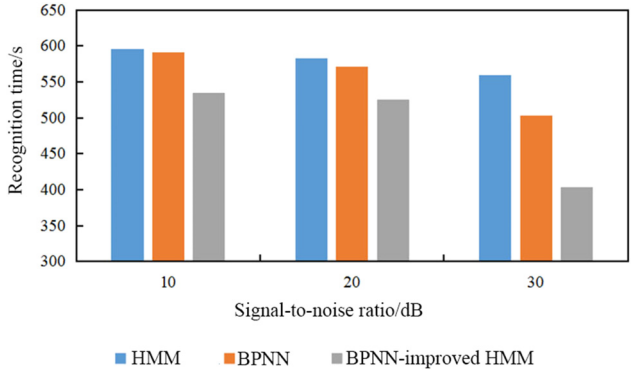

Figure 6: Recognition time consumption of three speech recognition algorithms under different signal-to-noise ratios.

4 Discussion

The recognition of English speech can be considered an important part of intelligent computer interaction. The computer itself cannot understand speech directly and needs to process it before recognition. Some studies on speech recognition are as follows. Aizawa et al. [21] studied a many-to-many voice conversion technology, whose basic principle is to recognize voice with HMM and regard the recognized phoneme sequence as the label of speech synthesis. The final experimental results showed that the system could effectively realize voice conversion. Lee and Jean [22] put forward a high-order hidden Markov model for piecewise linear processes and a parameter estimation method based on the expectation-maximization algorithm and found that the method could reduce the recognition error rate compared to the baseline hidden Markov model. This article recognized English speech with HMM and improved HMM with the BPNN algorithm to make up for the disadvantage of weak big data classification performance. Then, simulation experiments were performed on the HMM, BPNN, and improved HMM algorithms. With the increase of test samples, the word error rate of the three speech recognition algorithms gradually decreased, and the recognition time increased. The reason for the decreased word error rate is that the increase in the number of test samples led to an increase in the number of correct recognition. Why the decrease of the word error rate tended to be stable is because the number of samples wrongly recognized increased. The reason for the increased recognition time is that the increase in the number of test samples increased the burden of the algorithms, and the number of samples that could be recognized by the algorithms in unit time was limited.

Under the same number of samples, the improved HMM algorithm consumed less time in recognition than the other two algorithms. The reasons are as follows. The improved HMM algorithm rapidly extracted speech features with HMM and recognized features with BPNN that have advantages in parallel processing

of big data. The HMM algorithm, despite its advantages in fast modeling, was not good at the classification and the identification of big data. Although the BPNN algorithm could process big data in parallel, it did not have the ability to extract speech features. The improved HMM algorithm combined the advantages of BPNN and HMM algorithms, so that it could rapidly obtain accurate speech features and process a large amount of data in parallel to realize rapid classification and recognition of speech.

Under the interference of Gaussian white noise, the word error rate and recognition time of the three algorithms decreased as the noise interference level decreased. The reason why the word error rate increased is that the existence of noise interfered with speech features. The reason for the increased recognition time is that the speech features mixed with the noise made recognition more difficult. As described earlier, the improved HMM algorithm used a BPNN to classify and recognize speech after extracting speech features with HMM. BPNN not only was good at processing big data in parallel but also could mine the hidden rules from data; thus, it could reduce the influence of interference in speech features mixed with noise. Therefore, the word error rate and recognition time of the improved HMM algorithm were the lowest.

5 Conclusion

This article briefly introduced the HMM-based English speech recognition algorithm, combined it with the BPNN algorithm to further improve the recognition accuracy and reduce the recognition time, simulated the BPNN-combined HMM algorithm, and compared it with HMM and BPNN algorithms. The results are as follows. With the increase in the number of test samples, the word error rate of the three speech recognition algorithms decreased gradually, and the recognition time increased. Under the same number of test samples, the HMM algorithm had the highest word error rate and longest recognition time, followed by the BPNN algorithm and the BPNN-improved HMM algorithm. As the signal-to-noise ratio of test samples increased, the word error rate and recognition time of all the three recognition algorithms for speech recognition decreased. Under the same signal-to-noise ratio, the HMM had the highest word error rate and longest recognition time for speech recognition, followed by the BPNN algorithm and the BPNN-improved HMM algorithm.

This article improved HMM with the BPNN algorithm. The improved recognition algorithm used HMM to model speech quickly and extract features, while the BPNN algorithm processed the extracted features in parallel to quickly classify and recognize English speech. The subsequent experiments verified the effectiveness of the improved algorithm, providing an effective reference for intelligent computer recognition of English speech. The limitation of this article is that the speech samples used were mostly pure samples and were only added with white noise to test the anti-interference ability of the algorithm, but the recognition performance of the speech recognition algorithm for samples with unclear pronunciation is not tested. The future research direction is to improve the recognition performance of the speech algorithm for speech with unclear pronunciation.

Conflict of interest: The author declares no conflict of interest.

References

[1] Xiong W, Droppo J, Huang X, Seide F, Seltzer M, Stolcke A, et al. Achieving human parity in conversational speech recognition. IEEE/ACM T Audio Spe. 2016;99.

[2] Suominen H, Zhou L, Hanlen L, Ferraro G. Benchmarking clinical speech recognition and information extraction: new data, methods, and evaluations. JMIR Med Inf. 2015;3(2):e19.

[3] Saon G, Kuo H, Rennie S, Picheny M. The IBM 2015 English conversational telephone speech recognition system. Eurasip J Adv Sig Pr. 20082015;1:1–15.
[4] Kim C, Stern R. Power-normalized cepstral coefficients (PNCC) for robust speech recognition. IEEE/ACM T Audio Spe. 2016;24(7):1315–29.
[5] Watanabe S, Hori T, Kim S, Hershey JR, Hayashi T. Hybrid CTC/attention architecture for end-to-end speech recognition. IEEE J-STSP. 2017;11(8):1240–53.
[6] Sun S, Zhang B, Xie L, Zhang Y. An unsupervised deep domain adaptation approach for robust speech recognition. Neurocomputing. 2017;257(sep.27):79–87.
[7] Bhatt S, Jain A, Dev A. Monophone-based connected word Hindi speech recognition improvement. Sādhanā. 2021;46(2):1–17.
[8] Yavuz E, Topuz V. A phoneme-based approach for eliminating out-of-vocabulary problem of Turkish speech recognition using Hidden Markov Model. Int J Computer Syst Sci Eng. 2018;33(6):429–45.
[9] Lee LM, Le HH, Jean FR. Improved hidden Markov model adaptation method for reduced frame rate speech recognition. Electron Lett. 2017;53(14):962–4.
[10] Veisi H, Mani AH. Persian speech recognition using deep learning. Int J Speech Technol. 2020;23(4):893–905.
[11] Kuanyshbay D, Amirgaliyev Y, Baimuratov O. Development of automatic speech recognition for Kazakh language using transfer learning. Int J Adv Trend Comput Sci Eng. 2020;9(4):5880–6.
[12] Këpuska VZ, Elharati HA. Robust speech recognition system using conventional and hybrid features of MFCC, LPCC, PLP, RASTA-PLP and hidden Markov Model classifier in noisy conditions. J Comput Commun. 2015;3(6):1–9.
[13] Sun X, Yang Q, Liu S, Yuan X. Improving low-resource speech recognition based on improved NN-HMM structures. IEEE Access. 2020;8:73005–14.
[14] Zealouk O, Satori H, Laaidi N, Hamidi. M. Noise effect on Amazigh digits in speech recognition system. Int J Speech Technol. 2020;23(4):885–92.
[15] Kayte SN. Marathi speech recognition system using hidden Markov model toolkit. Concurrent Eng Res A. 2015;5(12):13–7.
[16] Chung Y. Vector Taylor series based model adaptation using noisy speech trained hidden Markov models. Pattern Recogn Lett. 2016;75(May 1):36–40.
[17] Vignesh G, Ganesh SS. Tamil speech recognizer using hidden markov model for question answering system of railways. Adv Intell Syst Comput. 2015;325:855–62.
[18] Khonglah BK, Dey A, Prasanna S. Speech enhancement using source information for phoneme recognition of speech with background music. Circ Syst Signal Pr. 2019;38(2):643–63.
[19] Awata S, Sako S, Kitamura T. Vowel duration dependent hidden Markov model for automatic lyrics recognition. Acoust Soc Am J. 2016;140(4):3427.
[20] Li K, Wang X, Xu Y, Wang J. Lane changing intention recognition based on speech recognition models. Transport Res C-Emer. 2016;69(Aug):497–514.
[21] Aizawa Y, Kato M, Kosaka T. Many-to-many voice conversion using hidden Markov model-based speech recognition and synthesis. J Acoust Soc Am. 2016;140(4):2964–5.
[22] Lee LM, Jean FR. High-order hidden Markov model for piecewise linear processes and applications to speech recognition. J Acoust Soc Am. 2016;140(2):EL204–10.