Research on the Retrieval Algorithm of Speech Keyword Based on DTW

Zhi Li, Zhiyuan Zhu and Jing Guo*

College of Electronic and Information Engineering Southwest University Chongqing, China

*Corresponding author: poem24@swu.edu.cn

Abstract. According to the query items entered by user, the speech retrieval is a data mining technology to retrieve and return the necessary results form massive audio data. In this paper, the history of speech retrieval is summarized and its necessity of studying is analyzed. On this basis, starting from the search of key words, the dynamic time regulation is derived in detail, and it is simulated by MATLAB. According to the results of MATLAB performance analysis, the optimization direction of speech keyword retrieval algorithm is explored.

1. Introduction

Currently, most of the information is converted into text data first, then the text data is searched and compared through computer to obtain the data we want. Although computer has a 100% accuracy for text query, most of the information in our life is composed of speech data in this era. It is time- and cost-consuming to convert all speech data into words and then build a database, which greatly thwarts the development.

In order to meet the increasingly growing needs of the speech information retrieval, it is necessary for us to study the speech retrieval algorithm, the basis of which is the key word retrieval. In the foreseeable future, speech retrieval is the trend of development. On April 11, 2006, Google submitted a patent application for speech technology to the US Patent Office, which involved a large number of technological inventions in speech recognition [1].

In addition, the reasons why I believe speech retrieval is a trend in the future are as follows: firstly, there is no need to convert new speech data into text, which saves time and is easy to operate; secondly, speech retrieval increases the amount of information recorded, and speech information can accurately record emotional information, while text is much worse in this aspect; thirdly, speech retrieval is closely connected with the artificial neural network [2] and can meet the development trend of the increasingly mature cloud computing technology.

Speech recognition technology and information retrieval technology [3] are two indispensable parts for the development of speech retrieval technology. Speech recognition is generally referred as automatic speech recognition [4] and information retrieval [5] is widely adopted by users to query and obtain information. As early as the 1950s, a speech recognition system that can recognize 10 English numbers was created in Bell Laboratories in the United States [6].

After decades of development, speech retrieval methods nowadays can roughly be divided into two categories: one is to convert speech to text, and then matching search is conducted in database [7]. The other is to convert speech into speech, that is, directly record the speech signal as input, then match the speech signal to the database, and finally output the result.
A speech to speech keyword retrieval algorithm based on feature matching of speech parameter \cite{8} is mainly studied in the paper.

2. Keyword retrieval algorithm based on dynamic time warping

The first step to study the speech to speech retrieval algorithm is from the matching of key words. At present, the method mainly adopted is to match the acoustic feature similarity \cite{9} between the input speech signal and the speech signal in the database and then find the position of the corresponding matching speech segment in the whole segment. But to achieve this function, it is needed to cut the speech signal first, and then match the input speech signal with the cut signal.

The speech signal length of the same word will be different in the time series because people pause at different points, or they have different speed when they are talking. Consequently, speech signal has a higher possibility of randomness in time sequence. Therefore, speech signal will have totally different time length in different times and under different circumstances, even if the same person speaks the same sentence.

2.1. Basic principle of dynamic time warping

Dynamic Time Warping (DTW) \cite{10} \cite{11} is a very simple and effective method in the field of isolated word recognition. Based on the algorithm logic of Dynamic Programming \cite{12}, DTW has solved the problem of different length in speech template matching \cite{13}. DTW primarily uses a time warping function W(n) which meets certain conditions to describe the time correspondence between the signal to be tested and the reference signal and it excels in the speech recognition of isolated word.

To put it another way, two sequences representing the same meaning will have similar shapes as a whole in most cases. The reason why the shapes are similar but not exactly the same may be caused by the compression or extension of time, thus resulting in the differences in the horizontal direction. Therefore, before calculating the similarity between the two, it is required to stretch or compress one sequence so that the parts with the same shape of the two can appear at the same time point. DTW is a good way to achieve this alignment. However, it has a limitation on the comparison of two speech signals, which need to be aligned at both ends.

In this way, DTW can know the start and end points of both, so endpoint detection \cite{14} is usually adopted to complete this step. As a result, the method is often used in the recognition of isolated words only, because for a long speech signal, we can not design a good algorithm to recognize the end of the sentence.

Speech signal is a kind of non-stationary signal and will encounter such interference as noise, current sound, etc. in the process of actual use, therefore the recorded speech signal can not be directly used for feature extraction, and before feature extraction, some processing needs to be carried out in advance. Firstly, it is necessary to filter out the interference of some signals that are obviously not human voice, and then carry out feature extraction, so that the extracted signal features are the closest to the original sound. As shown in Figure 1, for an input, the first step is to filter out the non-human noise with a band-pass filter\textsuperscript{1}, which can avoid the calculation of invalid signals and reduce the time consumption. Then sampling, framing, windowing and pre-emphasis are carried out. The signal obtained at this step can basically meet the requirements, but as we discussed above, two ends must be aligned if we use DTW, so the end point detection must be carried out finally, thus completing a preprocessing. MFCC \cite{15} feature extraction is used for the speech signal that has been preprocessed, because compared with LPC \cite{16} feature extraction, the former is more suitable for Chinese speech recognition. Then DTW is adopted to compare the known standard template data, and finally the retrieval results are output.

\textsuperscript{1} According to Baidu search engine, generally speaking, the voice range of people is between 100Hz (bass) and 10000Hz (soprano).
Suppose there are two time sequences X and Y, with time length m and n respectively. That is:

\[ X = X_1, X_2, ..., X_i, ..., X_m, \]
\[ Y = Y_1, Y_2, ..., Y_j, ..., Y_n; \]

Suppose that the two sequences have been framed and windowed, and their subscripts are equal to the number of frames. If \( m=n \), then the two sequences are naturally aligned at both ends. But in general, \( m \neq n \), then a matrix of \( m \times n \) needs to be constructed to align. When calculating the distance between elements of the sequence, Bhattacharyya distance is adopted [17].

Now, the similarity calculation of the two sequences is equivalent to finding the shortest distance path P from the matrix. However, the choice of this path is conditional, which satisfies:

1. Have boundaries. No matter how the speed of a sentence changes, the order of its components does not change, so the path must start from the lower left corner to the upper right corner. That is:
\[ P_1 = (1,1) \text{ and } P_k = (m, n) \] (1)

2. Continuity. To ensure that each point in the sequence is calculated once, the sub-path can only be linked by the frame adjacent to itself, rather than jumping over this frame to match the next frame. That is:
\[ P_{t+1} - P_t \in \{(0,1), (1,0), (1,1)\} \] (2)

3. Monotonicity. For a time sequence, no time reversal is allowed. That is, the value of \( n \) and \( m \) increases singly.

Now, with the constraints, we can roughly conclude that this is a continuous, monotonically increasing path from the lower left to the upper right. If the cumulative distance of this path is \( S(i,j) \) and the current distance is \( D(i,j) \), then the cumulative distance of this path can be obtained as follows:

\[ S(i,j) = D(i,j) + \min\{S(i-1,j-1), S(i-1,j), S(i,j-1)\} \] (3)

2.2. MATLAB implementation of dynamic time warping (DTW) algorithm

After understanding the principle and implementation process of DTW algorithm, we need to explore its feasibility on the program. Now we use MATLAB for experimental simulation.

Enter the two sequences we want to compare in the main function to find the shortest path. For example, if the input characteristic matrix is as follows:

\[ a=[8 \ 9 \ 1 \ 9 \ 6 \ 1 \ 3 \ 5]'; \]
\[ b=[2 \ 5 \ 4 \ 6 \ 7 \ 8 \ 3 \ 7 \ 7 \ 2]'; \]

Through computer calculation, the path of backtracking and the signal after DTW algorithm can be obtained as follows.
In this algorithm, I use two input eigenvectors to replace the input signal after feature extraction and the comparison signal in the feature template library. After that, starting from the upper left corner, I calculate the distance of the three adjacent paths (right, bottom, and bottom right), repeat the calculation until the end point, and finally find the shortest path by following the shortest distance.

As can be seen from the figure, after DTW algorithm, the computer can find out the position of the signal in the template more accurately.

Above we have simulated the matching of isolated words. What if we relax the endpoint detection and apply it to long speech signals? Next, we do not make any modification to the algorithm to simulate the retrieval of keywords in the long speech signal.

Run the program again and suppose that the two feature sequences are:

\[ a = [8 \ 9 \ 1 \ 9 \ 6 \ 1 \ 3 \ 5]'; \]
\[ b = [5 \ 4 \ 9 \ 2 \ 4 \ 8 \ 7 \ 3 \ 1 \ 0 \ 5 \ 1 \ 6 \ 3 \ 4 \ 2 \ 5 \ 4 \ 6 \ 7 \ 8 \ 3 \ 7 \ 2 \ 1 \ 2 \ 4 \ 8 \ 6 \ 3]'; \]

Note the underlined part, which is the feature template from the previous program.
template (circled). For example, 他是 (tashi) 一个踏实 (tashi) 的人 (He is a down-to-earth person). DTW algorithm may match the keyword “踏实 (tashi)” to “他是 (tashi)” and ignore the latter.

2.3. Optimization of DTW algorithm
As we have seen above, DTW algorithm is highly applicable to the matching of isolated words, but there will be problems once it is put into the retrieval of long speech. We still need to discuss its disadvantages in search of further optimization.

2.3.1. Advantages and disadvantages of DTW algorithm
First, let’s talk about the advantages. In my opinion, the greatest advantage of DTW algorithm is its high accuracy in the retrieval of isolated words. In addition, the retrieval of small number of data has low time complexity, small amount of memory consumption and low platform requirements.

Then let’s look at the disadvantages. Obviously, DTW is a matching algorithm for isolated words, and when we apply it directly to the retrieval of long speech signals, its effect becomes minimal. Furthermore, since the DTW algorithm requires both ends to be aligned, we need a better endpoint detection method to improve the calibration rate. Moreover, this algorithm needs to calculate a large number of paths and nodes, which leads to a huge amount of calculation. When the vocabulary in the template library is huge, the time cost will also increase greatly.

2.3.2. Optimization of DTW algorithm
As a retrieval algorithm of isolated words with high accuracy, DTW algorithm can not be used in the retrieval of long speech signals. So I wonder whether it can be improved for the retrieval of long speech signals. Here, I propose a possible way, that is, segment matching.

There is a keyword speech signal and a segment of speech signal, as shown in Fig. 4:

![Fig. 4 The schematic diagram of signal](image)

We first obtain the time length t of the keyword speech signal, then take t as the unit to truncate the continuous speech signal (usually, the time length of a keyword must be shorter than the time length of a paragraph) into n parts, and finally divide the remaining part less than t into one part separately. Then the keyword signal is matched with each part of the signal, and the similarity is calculated.

We usually speak at a relatively fast speed in real life, and the time length t of keywords’ speech signals emitted during retrieval will be longer than the time length of keywords in continuous speech. As a result, there are some possible matches in the truncated sequences, as shown in fig. 5. Generally speaking, our keywords will not be included in one truncated speech signal, so we will get multiple speech segments that match well with the head or the tail of the keyword.

![Fig. 5 The schematic diagram of segment matching](image)

Then, the signal nh matching with the head is added to the next signal nh+1, or the signal nt matching with the tail is added to the previous signal nt-1, to calculate the similarity again, as shown in Fig. 6, to compare the similarity between keywords and templates. In this way, we can determine
whether this part of the signal is a keyword, and then give the possible distribution of the target keywords in the continuous speech signal. Of course, this method assumes that the sequence length $x$ of the keyword is less than the sequence length $y$ of the suspected keyword contained in $2n$. But for now, except for a few exceptions, other cases are satisfied.

![Fig. 6 The schematic diagram of segment matching](image)

To summarize, suppose that under ideal conditions, the similarity between the feature templates of the keywords to be examined and the eigenvectors extracted from the keywords we speak is between 90% and 100%. When using this method, we will not only count the part that the similarity reaches the standard, but also set a threshold, like 50%. We add up the segments that reach this value with the preceding segment or the latter and calculate the similarity again. If the segment reaches the standard, it will be identified as a keyword. If not, it will be omitted.

In addition, the retrieval method proposed above has the time complexity similar to that of DTW algorithm and inherits other advantages of DTW algorithm. Therefore, in the case of networking, we can use cloud computing with higher computing efficiency to reduce the retrieval time.

3. Summary

In this paper, we summarize that text retrieval cannot meet the needs of modern information retrieval and point out that voice retrieval is the main direction of future development. According to the classification of speech retrieval methods, the basic principle of dynamic time warping (DTW) algorithm that directly uses speech signal to retrieve is first studied, and its implementation steps are listed. Experimental results are obtained through simulation, and its advantages and disadvantages are explored. In view of the fact that DTW cannot be used in long speech signal retrieval, an improved method called segment matching method is proposed. Then the principle of segment matching is explained and the possibility of its application is discussed.

At the same time, there are some shortcomings in this paper. For example, we did not consider the influence of signal truncation in DTW algorithm on the signal. Our default is that we can truncate the entire paragraph of the required keywords properly. In practice, the retrieval algorithm may not be able to retrieve the target due to inaccurate truncation. In segment matching, due to the randomness of truncation points, many useless calculations are caused and time complexity is increased. It is hoped that further research will be carried out in the future to overcome these shortcomings.

Acknowledgments
The research is supported by Fundamental Research Funds for the Central Universities (Grant No. SWU019040)

References
[1] Jie, T. Research on Content-based Audio Retrieval Technology.2010.
[2] Franzini,M.,A.,Lee,K.F., “A New Hybrid Method for Continuous Speech Recognition”, ICASSP-90,1990.
[3] Dong, L. Research on speech to speech retrieval system.2011.
[4] X.Huang,A.Acero,H.Hon. Spoken language processing: a guide to theory, algorithm and system development. s.l. : Prentice Hall Press,2001.
[5] MS lew,N Sebe,C Djeraba,R Jain. Content-based multimedia information retrieval: State of the art and challenges. s.l. : ACM New York,NY,USA,2006.Vols.ACM Transactions on Multimedia Computing, Communications, and Applications(TOMCCAP).1551-6857.
[6] Christiansen, & Rushforth, R.(1977). Detecting and Locating Key Words in Continuous Speech
Using Linear Predictive Coding. IEEE Trans. on ASSP. 25(5): 361-367.

[7] Foote, Jonathan. An Overview Of Audio Information Retrieval. s.l. : ACM Multimedia, 1999.

[8] Robine Matthias, Hanna Pierre, Ferrarro Pascal. Music similarity: improvements of edit-based algorithms by considering music theory. Proceedings of the international workshop on Workshop on multimedia information retrieval. 2007:135-142.

[9] Ney H. The Use of a One-stage Dynamic Programming Algorithm for Connected Word Recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1984, 32(2):263.

[10] Y. Zhu, D. Shasha. Warping Indexes with envelope transforms for query by humming. In Proceedings of the 2003 ACM SIGMOD international conference on Management of data, 2003:419-429.

[11] Duc.T.N.T, Nhat.M, Hoang.H.N.D, et al. Continuous pitch contour as an improvement feature for music information retrieval by humming/singing. In Proceedings of the 10th Pacific Rim International Conference on Artificial Intelligence, 2008:1086-1091.

[12] Vintsjuk, T.K, “Recognition of Words of Oral Speech by Dynamic Programming”, Kibernetica, Vol.81, No.8, 1968.

[13] Mingzhe, H. Research and application of speech segment retrieval algorithm. 2012:7-18.

[14] Wang H, Xu Y, Li M. IEEE International Conference on Artificial Intelligence, Management Science and Electronic Commerce. 2011.

[15] Davis S B, Mermelstein P. Comparison of parametric representations of monosyllabic word recognition in continuously spoken sentences. IEEE Transactions on Speech Acoustic Processing, 1980, 28:357-366.

[16] Witbrock, M. J., & Hauptmann, A. G. (1997). Using words and phonetic strings for efficient information retrieval from imperfectly transcribed spoken documents. In proceedings of the second ACM international conference on Digital libraries, ACM, 30-35.

[17] Deza, M. M. and Deza, E. Encyclopedia of distances. Heidelberg: Springer, 2009.