

Physician-patient speech separation method based on voiceprint technology and privacy protection

Li Zhang^{1,2,3*}, JingRui Liu² and Ming Jing^{1,2,3}

¹Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China

²Department of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China

³Shandong Provincial Key Laboratory of Computer Networks, Shandong Fundamental Research Center for Computer Science, Jinan, China

Abstract. In speech recognition and natural language processing of doctor-patient voice communication, it is critical to distinguish what comes from the healthcare worker and what comes from the patient. In addition, speech contains acoustic and linguistic features that can be identified by machine learning models to measure the speaker's behavioral health. At the same time, it is relatively simple and attractive for patients to use voice data acquisition, as well as relatively cheap and convenient, requiring only a microphone, a quiet place and a device to collect audio samples. Thus, voice-based biomarkers can pre-screen for disease, monitor disease progression and response to treatment, and be useful alternative markers for clinical studies with informed consent, but the premise of this process again requires us to distinguish between doctors and patients when taking audio samples. For audio samples of patient voices, in practice, most of the doctor's and patient's voices are not taken separately, but are mixed together. Several speaker recording methods have been used to isolate sound in the time domain; However, these studies do not address how to obtain timelabel-based speech samples, nor how to identify speakers. In this paper, a speech separation method is proposed for the audio separation situation between a doctor and several patients. The method mainly includes three parts: voiceprint segmentation clustering, cutting and splicing, speech identity determination. Doctor and patient audio can be separated while respecting the privacy of the conversation content, and can be stored separately based on the identity of the voice.

1. Introduction

In speech recognition and natural language processing of doctor-patient audio communication, it is crucial to distinguish which content comes from the medical staff and which comes from the patient. For example, the patient's description may not be accurate enough due to lack of professional medical knowledge; And the final treatment plan must be made by the doctor. Differentiating the voices of medical staff and patients through voiceprint segmentation clustering can help us better generate structured information for improving electronic medical records or automated diagnostic analysis.

In addition, speech contains acoustic and linguistic features that can be recognized by machine learning models to measure the speaker's behavioral health. At the same time, it is relatively simple and attractive for patients to use voice data collection, as well as relatively cheap and convenient, requiring only a microphone, a quiet place and a device to collect audio samples. Thus, voice-based biomarkers can pre-screen for disease, monitor disease progression and response to treatment, and be useful alternative markers for clinical studies with

informed consent, but the premise of this process again requires us to distinguish between doctors and patients when taking audio samples. And for patients who need privacy protection, it is also necessary to get the patient's voice feature data without knowing the content of the conversation.

For the audio samples of the patient's voice, in the actual situation, most of the doctor's and the patient's voice are not collected separately, but mixed together. Some speaker logging methods have been used to isolate sounds in the time domain; however, these studies do not address how to obtain speech samples based on temporal labels, nor do they address how to identify speakers.

The problem scenario to be solved in this paper is to separate the audio files kept by the same doctor in the dialogue with different patients through voiceprint technology and save the audio files to the corresponding files. Corresponding to this scenario, this paper proposes a voice separation method for one doctor and multiple patients based on voice print, which is mainly divided into three stages. In the first stage, the original audio is obtained through the voice print segmentation clustering model to obtain the corresponding RTTM file; In the second stage, according to the corresponding RTTM file,

* Corresponding author: lizhang@qlu.edu.cn

the source audio file is divided into different voice fragments of single speakers, and then spliced into complete sub-audio files according to the speaker information. The third stage determines the identity according to the number of times the speaker appears in the sub-voice file, and finally saves the audio of the doctor and each patient separately to the corresponding file.

2. Related work

Voiceprint segmentation clustering is also called speaker diarization, "Diarize" refers to the act of recording or writing down an experience in a diary. Speaker diarization is a method of addressing the problem of "who was speaking at the phase "by documenting speaker-specific salient events on multi-speaker audio records, analogous to keeping a diary [1,2,3].

Early speaker segmentation clustering technology is mainly used to improve the accuracy of automatic speech recognition. The methods proposed in the early stage mainly include Bayesian information criterion [4] and generalized likelihood ratio [5], which became the gold standard for measuring the similarity of speech fragments at that time. In the later development, some excellent methods emerged, such as beamforming [6], information bottleneck clustering [7], variable DB Bayesian [8], joint factor analysis [9] and so on. Later, i-vector [10] optimized according to JFA achieved great success and became the mainstream method of speaker classification system at that time. i-vector combined with principal component analysis and variable decibel Bayesian Gaussian mixed model made great progress compared with previous methods. After the outbreak of deep learning in 2021, d-vector [11,12,13] and x-vector [14] were put forward accordingly, and then, the end-to-end [15,16] training network gained attention. He replaced each submodule in the traditional speaker log with a neural network, and jointly optimized the parameters of the whole network through a loss function.

3. Speech separation framework based on voiceprint technology

The method of audio separation between a doctor and multiple patients is proposed in this paper, which mainly includes the following three parts. The frame diagram is shown in Figure 1.

-The RTTM file is obtained by inputting multiple interview source audio from different patients with the same doctor into the voiceprint segmentation clustering model.

-The original audio is cut and spliced according to RTTM, and the sub-separated audio of each audio file is obtained.

-The above sub-separated audio is segmented by voiceprint clustering, and the one that appears multiple times is identified as a doctor, and the one that appears once is identified as a patient. The doctor audio is spliced, and finally the separated audio of all patients and doctors

is obtained. The audio is named after the identity plus the speaker label of the last cluster.

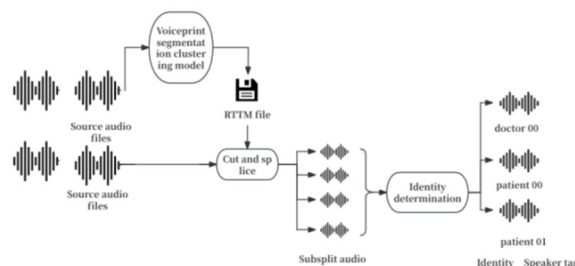


Fig. 1. Speech separation frame diagram

4. Voiceprint segmentation clustering method

The main content of this chapter is to obtain the corresponding RTTM files from the dialogue audio files between doctors and different patients through the voiceprint segmentation clustering technology.

4.1. Voiceprint segmentation clustering process

Voice print segmentation clustering generally goes through feature extraction, and then through speech activity detection, and can also go through speech overlap detection and speaker conversion detection, and then calculate the voiceprint embedding feature of each segment audio, and then cluster, so as to obtain the final annotation result. The voiceprint extraction model can be used to calculate the embedded features of voiceprint. Among them, the clustering model is a very important part of all the pronunciation separation methods. Hierarchical clustering, spectral clustering, and k-means clustering are often used. After the end of clustering, we can also take a new segmentation method to make the result more accurate. The resulting time stamp sequence is usually saved in RTTM format. The general flow chart of voiceprint segmentation is shown in Figure 2.

Based on our problem scenario, the number of voice and audio speakers is relatively fixed and there are few overlapping parts, so the requirements for the voiceprint segmentation clustering model are relatively low. At the same time, due to the consideration of computing resources and processing speed, the voiceprint segmentation clustering method proposed in this chapter conforms to the traditional architecture and has high stability. The speech activity detection part adopts mixed Gaussian model, speaker transformation detection based on left and right window comparison, speaker embedding adopts i-vector based on joint factor analysis, and cluster analysis adopts k-means clustering algorithm based on K-Means ++. Finally, a simple classification based quadratic segmentation is carried out. The details of the algorithms and techniques used in each section are described below.

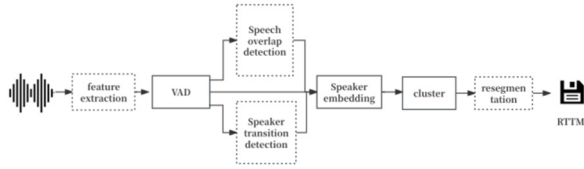


Fig. 2. General flow chart of voiceprint segmentation clustering

4.2. Voice Activity Detection

Using the speech detection model, the audio frame is divided into two categories: speech and non-speech. The non-speech class may be pure silent signals, or it may include ambient noise or other signals such as music or sound effects. Gaussian mixture model (GMM-VAD) was used to detect speech activity. The formula of Gaussian mixture model is shown in formula (1). The following is the formula and principle of Gaussian mixture model for speech detection:

$$p(x) = w_1 \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}} + w_2 \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{(x-\mu_2)^2}{2\sigma_2^2}} \quad (1)$$

Where, w_1 is the weight of the speech signal; w_2 is the weight of the non-speech signal; μ_1 is the mean value of the speech signal; σ_1^2 is the variance of the speech signal; μ_2 is the mean value of non-speech signals; σ_2^2 is the variance of the non-speech signal.

The principle of Gaussian mixture model for speech detection is to assume that both speech and non-speech conform to Gaussian distribution, and assume that non-speech is smoother than speech, and the energy is less than speech. That is, the mean variance of non-speech signals is smaller than that of speech signals. According to the formula, the GMM-VAD will have 6 parameters: the mean, variance, and weight of the speech signal, and the mean, variance, and weight of the non-speech signal. The signal is divided into 6 frequency bands: 80Hz~250Hz, 250Hz~500Hz, 500Hz~1KHz, 1KHz~2KHz, 2KHz~3KHz, 3KHz~4KHz. GMM is used to fit the signal for each frequency band. During the initialization of the GMM-VAD object, the above 6 parameters will use the default values. When the signal comes, the similarity probability is calculated according to the current GMM model and the current frame is judged as speech or noise. Then, according to the judgment, the GMM model is updated by using maximum likelihood estimation to update the above 6 parameters.

4.3. Voiceprint segmentation clustering process

After the completion of speech detection, the audio containing the speech signal needs to be segmented into many fragments in order to calculate the voicing embedding code and perform clustering. The shorter the segment, the less information it contains; The longer the segment, the more likely it is to contain multiple speakers. Through speaker conversion detection, the specific time point of speaker change is detected, so that the

segmented fragments meet the above two conditions at the same time.

Speaker conversion detection is based on left-right window comparison:

-For each audio frame, select the left and right Windows of the same size.

-The two Windows are then compared to see if the speech they contain comes from the same speaker. The specific comparison method is to calculate the voiceprint embedding code of the left and right Windows first, and then calculate the cosine similarity of the two embedding codes.

-Through the above method, a 2-dimensional curve can be obtained with the time of the audio frame as the X-axis and the cosine similarity of the left and right window voiceprint embedding code as the Y-axis.

-We first perform peak detection on the curve, that is, select all the local minima points, where the "local" can be selected with a specific length of window. Then all the local minimum points are compared with a specific threshold value. If the cosine similarity corresponding to the point is lower than the threshold value, it is considered that the left and right Windows of the audio frame belong to different speakers, that is, the audio frame has speaker conversion.

The detection density of the frequency frame is set to 100 milliseconds, the size of the left and right Windows is 2 seconds, and the peak detection window is 1 second.

4.4. Voiceprint embedding

After the speech segmentation is completed, the next step is to calculate the voiceprint embed code for each segmented segment. i-vector based on factor analysis is used in voicing code calculation, which is an improvement or simplification of joint factor analysis technique. Given a speech by a speaker, the Gaussian mean hypervector corresponding to it is shown in formula (2):

$$s = m + T \omega \quad (2)$$

Where, s is the Gaussian mean hypervector of a given speaker's speech fragment; m is a hypervector independent of both speaker and channel. T is the global difference space matrix; ω is a random vector $\{x_i\}$ that follows a standard multivariate normal distribution, with dimensions usually chosen between 400 and 600. The overall factor ω is also known as an "identity vector", abbreviated as i-vector. By calculating the Baum-Welch statistic corresponding to the target speaker, the global difference space matrix T is estimated, and the posterior mean of ω is calculated as i-vector.

4.5. clustering analysis

After obtaining the voiceprint embed codes for each segment, the next step is to perform cluster analysis on these embed codes. In the cluster analysis, it is necessary to determine how many categories the voiceprint embeddings from these fragments belong to, and assign each voiceprint embeddings to a specific category. Each

of these categories corresponds to a speaker. After cluster analysis, we get the preliminary result of voiceprint segmentation clustering.

Clustering adopts K-means clustering algorithm, and K is 2. The clustering formula is shown in formula (3). The mission objectives are described as follows:

Divide the N then data $\{x_i\}_{1 \leq i \leq N}$ into k subsets $S = \{S_1, \dots, S_k\}$, such that the sum of squares within the class is minimized. μ_i represents the mean vector of all data in the subset S_i . And $d(\cdot)$ represents the cosine distance based on cosine similarity.

$$\arg \min_S \sum_{i=1}^k \sum_{x \in S_i} d(x, \mu_i)^2 \quad (3)$$

The cosine distance calculation formula based on cosine similarity is shown in formula (4).

$$d(x, \mu_i) = \frac{1 - \cos(x, \mu_i)}{2} \quad (4)$$

The algorithm is described below:

- Initialization. Select the center of the initial k classes.
- Allocate. For each data, its cosine similarity to the center of the k classes is calculated, and it is assigned to the class with the highest cosine similarity.
- Updated. For each class, based on the data assigned to the class, calculate the mean vector of these data as the new center of the class.
- Iterate. Repeat steps 2 to 3 above until the convergence criteria are met and the change in the sum of squares within the class is less than the threshold.
- For each audio frame, select the left and right Windows of the same size.

A good initialization strategy can greatly reduce the number of assignments and updates required for convergence. Therefore, the initialization strategy uses the K-Means++ method, the specific steps of which are as follows:

- The initial point. The center of the first class is randomly selected from all the data according to the uniform distribution.
- Calculate the distance. For each point in the data, calculate its distance to the center of the existing class and take the minimum value of these distances.
- Random sampling. A random point from all the data is chosen as the center of the next class, but the probability of the random distribution here is proportional to the square of the distance minimum calculated in step 2.
- Iterate. Repeat steps 2 to 3 above until all k centers are selected. As you can see from the above steps, the advantage of the K-Means++ method is that it can avoid simultaneously selecting points that are too close together as the center of the initial class, and can take care of points that are far away from most of the data.

4.6. Re-segmentation

After the cluster analysis is completed, the corresponding speaker has been obtained for each segment, but the result may still be not ideal. Secondary segmentation is a post-processing method after the completion of clustering,

which is used to further adjust the fragment boundaries and clustering results, so as to obtain better results.

Re-segmentation adopts the simplest classification based quadratic segmentation, that is, the results of the clustering algorithm are used as training data, a classifier model is trained, and then the classifier is applied to the same data to get classification results. We get the initial segmentation clustering result $\{(x_i, y_i)\}_{1 \leq i \leq N}$, that is, the fixed length of audio fragments corresponding to each speaker (there are N such fragments in total). Then, we can extract the features $\{z_i\}_{1 \leq i \leq M}$ from the more dense $M > N$ audio fragments, find the speaker $1 \leq i \leq M$ corresponding to these features, and thus train a two-class classifier. Finally, the trained classifier is applied to the M dense features $\{z_i\}$, and a new classification result is obtained. As the final result of the secondary segmentation, the corresponding start and end time and speaker label of each segment are saved in the RTTM file. The effect of the re-segmentation is shown in Figure 3.

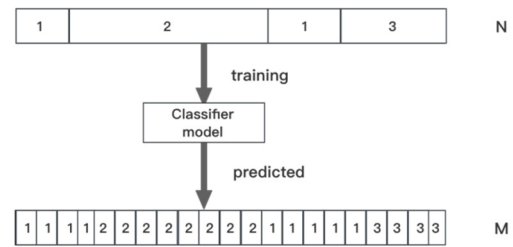


Fig. 3. Re-segmentation effect picture

5. Clipping and splicing method

Through the voiceprint segmentation clustering model in Chapter 4, we can obtain the RTTM file corresponding to each source audio. In this chapter, our task is to segment the source audio according to the speaker information in the RTTM file, and then splicing the voice fragments of the same speaker to obtain the single speaker audio subfile corresponding to each source audio file. The cutting and stitching method proposed in this chapter mainly includes the following five steps. The frame diagram of the cutting and splicing method is shown in Figure 4, and the flow diagram of the specific algorithm is shown in Figure 5.

- Iterate over the start time cut-off time and speaker label in the RTTM.
- A single start time and end time are stored in the circulation.
- Clip the source audio file according to the start time and end time, and save the temporary split file.
- Check whether there is an audio file named by the speaker label in the save directory. If no audio file is named by the speaker label, save the temporary split file. If an audio file named by the speaker label already exists, the splice command is executed to splice the temporary split file after the existing file.
- When the traversal is complete, you get two sub-audio files named after the speaker label.

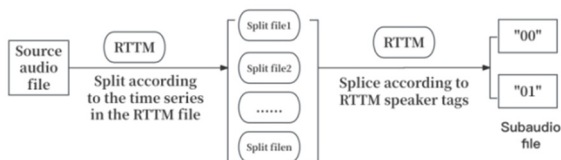


Fig. 4. Flow chart of clipping method

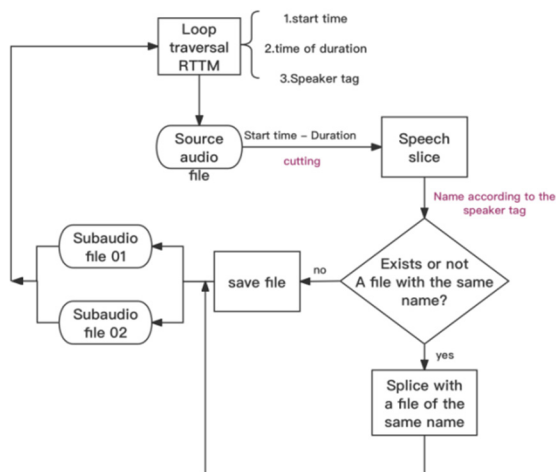


Fig. 5. Flow chart of clipping method

6. Speech identity determination method

The main task of this chapter is to determine the identity of the sub-separated audio obtained in Chapter 5. In the context set by this paper, there is a doctor with the same voiceprint information in each source audio file, so the method proposed in this chapter is to identify the above sub-separated audio by voiceprint again. In order to improve the recognition efficiency, we extract 3s audio from sub-audio files for voiceprint recognition. For the speaker who appears repeatedly, we identify him as a doctor. For the speaker who only appeared once, we determined it to be a patient. Finally, we spliced all the sub-separated audio of the doctor, and finally saved the audio of the doctor and the patient by naming the label of identity + speaker. The method mainly consists of the following five steps, and the flow chart is shown in Figure 6.

-The subaudio files generated by each source file are extracted into any 3-second continuous segment.

-Extract the voiceprint embed code. This step follows the same approach as in Chapter 4.

-By cluster analysis, speaker labels of each sub-separated audio are obtained. The k-means algorithm mentioned in Chapter 4 is also used for cluster analysis.

-Determine whether the speaker label of the subaudio is duplicated

-The subaudio files with repeated identities are spliced and named "Doctor + identification serial number", and the subaudio files whose identities appear only once are named "patient + identification serial number".

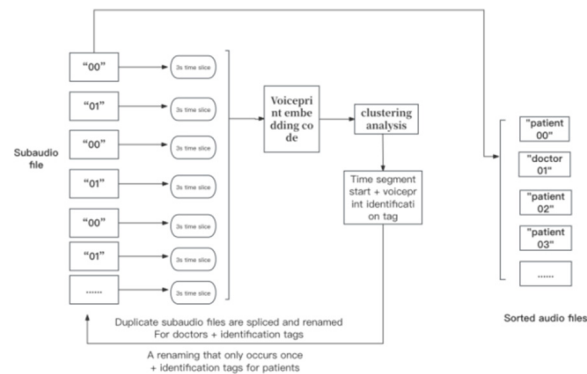


Fig. 6. The flow chart of voice identity determination method

7. conclusion and prospect

This article solves the problem of how to separate the audio of the conversation between the same doctor and different patients. The speech separation method provided in this paper mainly includes the input of multiple inquiry source audio of different patients corresponding to the same doctor into the voice print segmentation clustering model to obtain the RTTM file corresponding to each inquiry source audio; According to the RTTM file, the corresponding original audio is cut and spliced, and the sub-separated audio corresponding to each original audio is obtained. The sub-separated audio corresponding to the original audio of each consultation was identified by voiceprint, and the one that appeared multiple times was identified as the doctor's audio and the one that appeared once was identified as the patient's audio. The doctor audio is spliced to obtain multiple separated audio of different patients and doctors, and each separated audio is named after the identity and speaker label. This method can achieve the separation of the communication audio of doctors and patients, and obtain voice samples according to the time series label, and determine the identity information of the speaker.

The speech separation method proposed in this paper can improve the development of electronic medical records and automated diagnostics, while distinguishing between doctors and patients when collecting audio samples, and can also greatly improve the efficiency of speech data acquisition and speech-based biomarker studies. Another applicable scenario of this method is voice data extraction based on privacy protection, which can obtain the corresponding voice features of patients while respecting the privacy of the dialogue content of patients. But at the same time, the method proposed in this paper does not solve the problem of overlapping speech, There are great limitations to the applicable situation such as the method is not suitable for the situation of multiple doctors and multiple patients.

In the future, we hope to optimize the method so that it can adapt to more diverse and complex situations, and we intend to use the method to build a speech database of people with mental illness.

References

1. S. S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," in *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1557-1565, Sept. 2006, doi: 10.1109/TASL.2006.878256.
2. X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland and O. Vinyals, "Speaker Diarization: A Review of Recent Research," in *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356-370, Feb. 2012, doi: 10.1109/TASL.2011.2125954.
3. Tranter, Sue et al. "An Investigation into the Interactions between Speaker Diarisation Systems and Automatic Speech Transcription B Accuracy of Cts Forced Alignments 44." (2003).
4. H. Gish, M. . -H. Siu and R. Rohlicek, "Segregation of speakers for speech recognition and speaker identification," [Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing, Toronto, ON, Canada, 1991, pp. 873-876 vol.2, doi: 10.1109/ICASSP.1991.150477.
5. Chen, Scotte et al. "Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion." (1998).
6. X. Anguera, C. Wooters and J. Hernando, "Acoustic Beamforming for Speaker Diarization of Meetings," in *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2011-2022, Sept. 2007, doi: 10.1109/TASL.2007.902460.
7. D. Vijayasenan, F. Valente and H. Bourlard, "An Information Theoretic Approach to Speaker Diarization of Meeting Data," in *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 7, pp. 1382-1393, Sept. 2009, doi: 10.1109/TASL.2009.2015698.
8. F. Valente, P. Motlicek and D. Vijayasenan, "Variational Bayesian speaker diarization of meeting recordings," 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX, USA, 2010, pp. 4954-4957, doi: 10.1109/ICASSP.2010.5495087.
9. P. Kenny, D. Reynolds and F. Castaldo, "Diarization of Telephone Conversations Using Factor Analysis," in *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 6, pp. 1059-1070, Dec. 2010, doi: 10.1109/JSTSP.2010.2081790.
10. N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," in *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788-798, May 2011, doi: 10.1109/TASL.2010.2064307.
11. E. Variiani, X. Lei, E. McDermott, I. L. Moreno and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 2014, pp. 4052-4056, doi: 10.1109/ICASSP.2014.6854363.
12. G. Heigold, I. Moreno, S. Bengio and N. Shazeer, "End-to-end text-dependent speaker verification," 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 2016, pp. 5115-5119, doi: 10.1109/ICASSP.2016.7472652.
13. Q. Wang, C. Downey, L. Wan, P. A. Mansfield and I. L. Moreno, "Speaker Diarization with LSTM," 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 2018, pp. 5239-5243, doi: 10.1109/ICASSP.2018.8462628.clustering},
14. D. Snyder, D. Garcia-Romero, G. Sell, D. Povey and S. Khudanpur, "X-Vectors: Robust DNN Embeddings for Speaker Recognition," 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 2018, pp. 5329-5333, doi: 10.1109/ICASSP.2018.8461375.
15. Fujita, Yusuke, et al. "End-to-end neural speaker diarization with permutation-free objectives." arXiv preprint arXiv:1909.05952 (2019).
16. Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu and S. Watanabe, "End-to-End Neural Speaker Diarization with Self-Attention," 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Singapore, 2019, pp. 296-303, doi: 10.1109/ASRU46091.2019.9003959.