UDC 004.8

M. Drahan, A. Pysarenko

PREPROCESSING OF AUDIO DATA FOR VOICE TRANSCRIPTION SYSTEMS

Abstract: Voice messages are currently a powerful data collection tool. The aim of the study is to speed up the transcription of audio files. To achieve the goal, it is suggested to use a bandpass filter with a lower passband frequency in the range of 150-200 Hz and an upper passband frequency in the range of 3500-7000 Hz. The success of the system is based on the selection of the filter that optimally speeds up the transcription of audio files.

Keywords: audio file, transcription, low-pass filter, high-pass filter, bandpass filter.

Problem statement

Transcription of voice messages is becoming more and more relevant throughout the world and it is an important tool for data collection and analysis, particularly in the fields of science and research. Thus, by transcribing audio or video recordings, a significant amount of data can be collected in linguistics, sociology, political science, marketing, and many other fields. In addition, transcribing can become a primary communication tool for people with hearing impairments, allowing them to understand those around them more easily.

It is worth noting that transcribing can be useful for improving work efficiency in business and education. Recordings of webinars, conferences, meetings and interviews can be transcribed for further analysis and used for various purposes.

In the modern world, the vast majority of automatic text transcription methods are based on the use of neural networks. Despite their high efficiency, the main disadvantage of these methods is the use of large calculating power, which requires significant material costs. To solve this problem, the use of graphics processors was proposed, which allows to significantly speed up the calculation process compared to the central processors.

However, the use of automatic speech recognition systems requires the presence of a high-quality sound. This means that any noisy environment that people normally have to deal with can significantly slow down the system. Thus, to ensure the efficiency and accuracy of automatic transcription, it is necessary to ensure the proper quality of the input audio or video recording.

Analysis of recent research and publications

Increase of the playback speed of the audio can help speed up the transcription process. However, it is worth noting that automatic speech recognition (ASR) systems were trained on models of normal human speech. Therefore, a slight acceleration can reduce the file size and speed up the transcription process for human perception. However, with too much acceleration, neural networks can spend more time analyzing the audio track.

Filtering is not a universal approach to solving the problem of improving the quality of transcriptions. The sound frequency of different voices can differ significantly too, so filters with certain standard values are used. In addition, filters can deprive words of certain depth and clarity. For example, ASR-systems, in the event of ambiguity in the transcription of a word, try to check it again and choose the most likely option. With excessive filtering, the system performs additional operations to check the clarity of the sound of words, which can significantly slow down the transcription process.

Article [1] discovers the effect of different filtering and transcription methods on the results of speech recognition using ASR systems. The authors conducted an experiment using the ASR system based on DeepSpeech2 and data from a test set using several sound sources and with additional sound processing using various filtering and transcription methods. The article explores various filtering methods such as high-pass filter, low-pass filter, Kalman filter, and median filter. The results of the experiment showed that the use of a low-pass filter improves speech recognition results, while the use of a high-pass filter worsens the results. Therefore, the paper demonstrates the importance of applying effective filtering and transcription techniques to improve speech recognition results using ASR systems.

The article [2] examines the effect of sound pre-processing methods on automatic transcription of spoken speech. The authors conducted an experiment using different methods of noise reduction on data from spontaneous speech. For the study, the authors used the Must-C database, which contains spoken English and French texts recorded in different noise conditions. They compared the accuracy of speech recognition using automatic transcriptions with and without noise reduction using two different accuracy metrics: Word Error Rate (WER) and Character Error Rate (CER). The results of the study showed that the use of noise reduction methods significantly improves the accuracy of automatic transcription. The best result was obtained using the noise reduction method, which is based on the smoothing of spectral coefficients, as well as on the maximum posterior probability. Application of this method made it possible to reduce WER by 4.4% and CER by 4.6% compared to transcription without noise reduction.

Article [3] investigates the effect of applying sound filtering on the results of speech recognition in noisy conditions for listeners with normal hearing and hearing impairment. The study uses two noise reduction methods - SpeexDSP and Wavelet-AGC - and examines their effect on transcription quality compared to the original audio recordings. The study was conducted on the basis of a corpus of spoken language consisting of 10 hours of audio collected using a microphone at a distance of 30 cm from the conversation participant. The audio recordings contained various types of noise, such as background noise, ventilation noise, and keyboard noise. As a result of the study, it was found that both methods of noise reduction improve the quality of the transcription compared to the original audio recordings. Using the SpeexDSP method resulted in an improvement in speech recognition accuracy of 3.9% for low-noise audio recordings and 6.1% for high-noise audio recordings. Wavelet-AGC also improved speech recognition accuracy, especially for low-noise audio recordings.

In [4], the authors investigate the effect of using filtering and noise reduction algorithms on speech recognition using automatic speech recognition (ASR) systems. In the work, the authors used two ASR models: Hidden Markov Model (HMM) and Long Short-Term Memory (LSTM) to analyze the effect of various filters and noise reduction algorithms on speech recognition in noisy conditions. Multi-filter banks containing different filters such as Gaussian filter.

Kalman filter and noise filter were used for filtering. As a result of the experiments, the authors established that the use of multi-filter banks improves the results of speech recognition in noisy conditions. In addition, they proved that the use of noise reduction algorithms, such as adaptive noise filter and Wiener filter, can significantly improve the quality of speech recognition.

The article [5] is dedicated to the comparison of the effectiveness of two noise reduction algorithms - spectral subtraction and the Wiener filter for improving speech recognition in noisy conditions. In the paper, the authors conducted experiments on the TIMIT database, which included both clean and noisy sound files, and compared the speech recognition results of the two noise reduction algorithms. They also compared the performance of each algorithm at different levels of noise. As a result of the experiments, the authors concluded that both noise reduction algorithms gave a positive effect on speech recognition in noisy conditions, however, spectral subtraction was more effective at low noise levels, while the Wiener filter was more effective at high noise levels.

Complex algorithms described in articles [2, 3, 4] definitely increase the quality of the received text, but the goal of this article is to reduce the time required for transcribing the text. The article [1] considered the influence of such fast-filtering methods as the use of low-pass filter and high-pass filter, but the use of a band-pass filter was not investigated. This article is dedicated to this problem.

Main research material

In the frame of the research, the Whisper open-source code with the "large-v2" language model was used. Preliminary analysis showed that this model is the most appropriate in terms of speed and quality of the received transcribed text. To evaluate the effectiveness of the filters, three sets of audio data were generated:

– "pure sound" is the sound that is closest to the original, that is, it does not contain noise, interference, distortion, artifacts, etc. This sound seems almost perfect to a person;

– "partially noisy" – this sound contains noise components, but still retains some clarity and distinction between different sounds. The vast majority of words can be identified the first time, with the exception of a few. Sometimes there may be some random sounds;

– "noticeably noisy" – this sound contains noise through. Although the words can be recognized, they require additional effort. This sound can be quite unpleasant for human hearing.

First of all, data collection and separation of audio files into the three categories mentioned above is carried out. After that, to evaluate the quality of the impact of a particular filter, filter and transcribe the file.

Although these processes are performed sequentially, they can be significantly separated in time, so we will consider them separately. In the preliminary filtering of audio files, a self-developed Python application was used, using the following libraries:

– Time – Python language library that provides functions for working with time and dates and allows you to record processing time;

– Pydub – an audio library for trimming, merging, splitting and converting audio files. It is built over the basis of the ffmpeg library and can work with many audio file formats, such as mp3, wav, flac, ogg and others;

– IO – a library that provides the ability to work with input and output data streams, such as text files, binary files, strings, etc.

The specified set of libraries is sufficient for full filtering of an audio or video file. In fig. 1 shows a diagram of the preliminary preparation of the file for the transcription process.

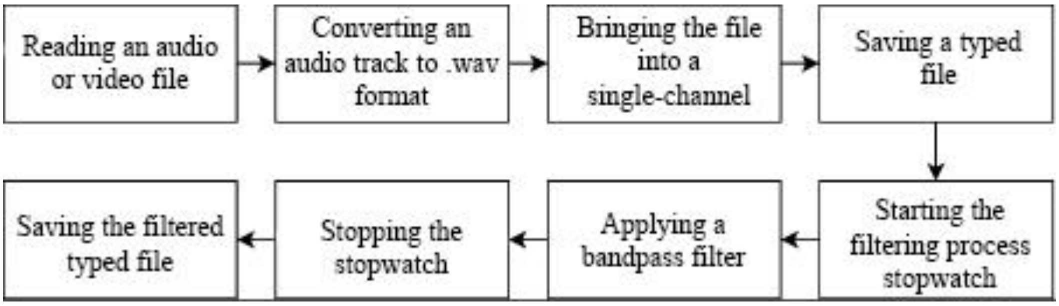
Reading an audio or video file
Converting an audio track to .wav format
Bringing the file into a single-channel
Saving a typed file
Starting the filtering process stopwatch
Applying a bandpass filter
Stopping the stopwatch
Saving the filtered typed file

Figure 1. Scheme of data preparation and filtering

Typically, audio tracks can be derived from audio files or from video files. Most ASR models use a standard single-channel WAV file to convert the audio file to the desired format. Further, the data is reduced to mono channel mode. The resulting mono-channel file is filtered using the specified filter parameters, after which the elapsed time is saved and the filtered file is written to disk. These operations are performed for each file in the three groups. After that, the data is subjected to the transcription process. For convenience and data reliability, the functions were divided into two logical blocks that can be executed in parallel. In fig. 2 shows the diagram of the transcription unit.

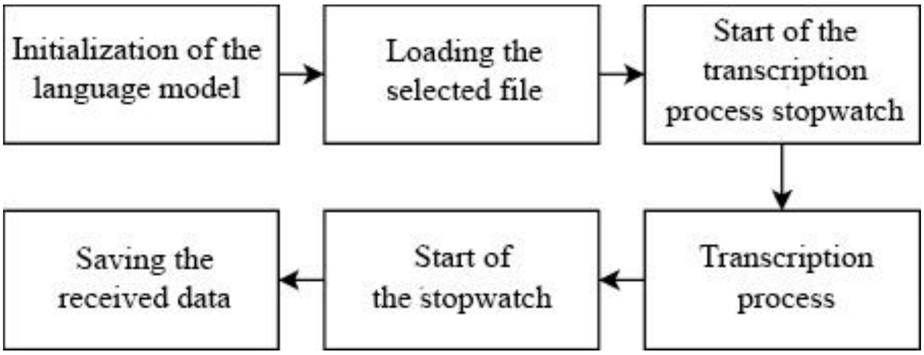
Initialization of the language model
Loading the selected file
Start of the transcription process stopwatch
Transcription process
Start of the stopwatch
Saving the received data

Figure 2. Scheme of files transcription

Whisper is an acoustic model that offers a variety of customization options. However, the recommended parameters to use are beam_size=5 and best_of=5. The beam_size parameter is used to adjust the size of the side panel of the acoustic model during speech recognition. This parameter affects the speed and accuracy of speech recognition. Generally, a larger beam_size value provides higher accuracy, but lowers the recognition speed. The best_of command is an option to the recognize command in the Whisper library. It indicates the number of best hypotheses returned for speech recognition. For example, if you specify best_of=3, the three most likely speech recognition hypotheses will be returned for each audio file. The large-v2 language model is used for testing, which provides an optimal ratio of transcription speed and quality.

After initialization, the Whisper system automatically buffers an audio file from the file system by its name. A stopwatch is started to measure the transcription time. The time and result of transcription are saved for further processing. After transcribing each file from the three groups, the resulting textual data is compared between the corresponding pairs. The study found that all pairs were more than 95% concordant, so transcription results before and after filtering were considered "the same" and could be compared over time. Formula (1) is used for the percentage comparison of the transcription speed of two files in a pair.

$$x = \left[1 - \frac{t_f + t_{ft}}{t_t}\right] * 100\% \ , \tag{1}$$

where t_f – time of prefiltering the file,

t_{ft} – the time of transcribing of the filtered file,

t_t – time to transcribe the original file.

Naturally, the higher the obtained indicator, the better the filtering result.

In experiments, all filters are Butterwort filters of order 6 and a gain of 1.

On the basis of the groups of audio data that were described above, a check of the quality of the high-pass filter (HPF) was carried out. The human voice can occupy frequency intervals from 20 Hz to 20 kHz. First, the filtering was performed from the lower limit of 50

Hz, and then the frequency values were gradually increased until the deterioration of the result was not noticed.

The filtering results are presented in the form of a percentage ratio of the transcription speed of the filtered files to the initial data shown in Table 1.

Table 1.

Results of HPF

Frequency of HPF, Hz	**Clear sound, %**	**Partly noisy, %**	**Noticeably noisy, %**
50	2	-17	29
100	6	5	35
150	3	5	23
200	-5	12	21
250	-8	-49	20

As can be seen from the table, in all cases there is an acceleration of the transcribing system. As mentioned above, filtering can increase the overall filtering time, for example in situations with pure and semi-pure sounds, at a frequency of 250 Hz.

A similar experiment was performed with a low-pass filter (LNP), the results of which are presented in Table 2.

Table 2.

Results of LPF

Frequency of LPF, Hz	Clear sound, %	Partly noisy, %	Noticeably noisy, %
3000	-3	-25	7
3500	-10	-19	13
4000	1	-24	33
4500	0	-18	7
5000	-5	8	22
5500	-1	-21	20
6000	-6	-20	41
6500	0	-13	7
7000	-3	-21	33

From the obtained results, it can be seen that the influence of low frequency is much less predictable than the influence of high frequency. This may be due to the specificity of neural network training. The key features of the words that the network uses to make decisions are significantly different from human perception. In addition, high-frequency noise is not usually found in everyday life, unlike low dull sounds. Therefore, the low-frequency filter, although it speeds up the operation of the system, in a situation of noticeable noise, spoils the result in other cases. Therefore, it is advisable to consider the operation of bandpass filters.

Table 3 presents the results of the bandpass filter with a lower frequency of 50 Hz and an upper frequency of the transmission band from 3000 Hz to 7000 Hz.

Table 3.

Results of the bandpass filter with a lower frequency of 50 Hz

The upper frequency of the bandwidth, Hz	Clear sound, %	Partly noisy, %	Noticeably noisy, %
3000	4	-2	21
3500	-38	-10	-5
4000	-1	-25	26
4500	2	-20	45
5000	-1	-19	34
5500	4	-20	1
6000	1	-20	4
6500	7	-22	23
7000	-2	-2	-12

Based on the results presented in the table, it can be concluded that at a lower frequency of 50 Hz and a bandwidth frequency of 4-5 kHz, the band-pass filter helps to improve the results.

The results of the bandpass filter with a lower passband frequency of 100 Hz are shown in Table 4.

Table 4.

Results of the bandpass filter with a lower passband frequency of 100 Hz

The upper frequency of the bandwidth, Hz	Pure sound, %	Partly noisy, %	Noticeably noisy, %
3000	3	-21	26
3500	-7	-21	25
4000	-5	13	24
4500	3	7	8
5000	-5	8	19
5500	-8	5	25
6000	4	-19	34
6500	-7	5	20
7000	2	-23	22

As can be seen from Table 4, the bandpass filter with a lower passband frequency of 100 Hz performed much better, as there are both purely positive lines and a noticeable acceleration in two of the three groups, with a slight deterioration in the third.

The next step is to consider the operation of a band-pass filter with a lower passband frequency of 150 Hz, the research results are shown in Table 5.

Table 5.

Results of the bandpass filter with a lower passband frequency of 150 Hz

The upper frequency of the bandwidth, Hz	Clear sound, %	Partly noisy, %	Noticeably noisy, %
3000	-7	-24	24
3500	0	12	31
4000	-4	3	8
4500	-6	7	24
5000	5	7	23
5500	-6	7	33
6000	-6	12	23
6500	-4	5	23
7000	-6	4	25

As can be seen from the results of the bandpass filter with the lower passband frequency of 150 Hz, a significant acceleration was obtained at the upper passband frequency of 3500 Hz and 5000 Hz. However, in general, it is worth noting that from the upper frequency of the passband 3500 Hz inclusive, the system produced a higher speed of operation in noisy cases, compared to the slowdown for clean sound. The availability of pure sound, given the widespread use of transcription systems, is unlikely. So, in a broad sense, a bandpass filter with a lower frequency of 150 Hz and an upper frequency of the passband of 3500-7000 Hz can currently be considered a universal solution.

Consider the effect of a bandpass filter with a lower passband frequency of 200 Hz (table 6).

As can be seen from the results obtained from the operation of the bandpass filter with a lower passband frequency of 200 Hz, the option of using the upper passband frequency of 3000 Hz can definitely be rejected, but all other options, especially the range of 4000-5000 Hz, showed very high results with significant noise, and not only did not slow down, but also accelerated the work of transcribing the other two groups.

And, finally, considering the results of the bandpass filter with a lower passband frequency of 250 Hz, shown in Table 7.

Table 6.

Results of the bandpass filter with a lower passband frequency of 200 Hz

The upper frequency of the bandwidth, Hz	Clear sound, %	Partly noisy, %	Noticeably noisy, %
3000	-41	2	8
3500	-6	7	7
4000	2	7	51
4500	4	7	25
5000	1	1	49
5500	-7	1	24
6000	-6	4	25
6500	-1	3	23
7000	-7	7	24

Table 7.

Results of the bandpass filter with a lower passband frequency of 250 Hz

The upper frequency of the bandwidth, Hz	Clear sound, %	Partly noisy, %	Noticeably noisy, %
3000	-6	9	23
3500	-5	-13	43
4000	-1	10	24
4500	3	-45	45
5000	-7	13	32
5500	-5	16	24
6000	-4	2	18
6500	-25	-11	24
7000	-18	1	25

As can be seen from the results of the work, the bandpass filter with the lower frequency of the passband of 250 Hz, as it was a dubious option when using only high frequency with a cut off frequency of 250 Hz, remained so.

Conclusion

As a result of pre-filtering groups of audio files, the advantages of using a bandpass filter with a lower passband frequency in the range of 150-200Hz and an upper passband frequency in the range of 3500-7000Hz were clearly demonstrated. The best options were definitely highlighted, namely: bandwidth range 150 - 3500 Hz, 150 - 5000 Hz, 200 - 4000 Hz, 200 - 4500 Hz, 200 - 5000 Hz. However, in general, the use of the specified filters in the described ranges allows you to speed up the process of transcribing the text. Thus, it is possible to achieve an increase in transcription speed not only due to the use of video cards, but also the use of central processors and pre-filtering. In the future, it is possible to consider in more detail the use of bandpass filters with a lower passband frequency in the range of 150-200 Hz and an upper passband frequency in the range of 3500-7000 Hz. In addition, consider the removal of empty segments based on the energy of the signal and the effect of voice acceleration on the time of its transcription, using the above-mentioned filters.

REFERENCES

1. Banu, A., & Shahin, S. Effectiveness of filtering and transcription method on speech recognition performance. International Journal of Speech Technology, 2018.
2. Litvinov, A., Metze, F., & Schatz, J. Impact of noise reduction on the automatic transcription of spontaneous speech. Speech Communication, 106, 46-56, 2019.
3. Burkhardt, D., Duduch, A., & Fitch, L. The Effect of Audio Filtering on Speech Recognition in Noise for Listeners with Normal Hearing and Hearing Impairment. Journal of the American Academy of Audiology, 28(2), 140-149, 2017.
4. Hong, Y., Lee, J., Kim, M., & Lee, J. Speech enhancement and recognition using filter banks and noise reduction algorithms. Journal of Electrical Engineering and Technology, 13(3), 1328-1339, 2018.
5. Kwon, O., Lee, C., & Lee, S. The effect of speech enhancement on speech recognition: A comparison of spectral subtraction and Wiener filtering. Applied Sciences, 8(12), 2609, 2018.