Journal of Computer Science

Original Research Paper

Speech Segmentation Using Dynamic Windows and Thresholds for Arabic and English Languages

Yahia Hasan Jazyah

ITC, Arab Open University, Kuwait

Article history
Received: 04-01-2018
Revised: 20-03-2018
Accepted: 17-04-2018

Email: yehia_hassan@yahoo.com

Abstract: Segmentation of audio data such as human speech (splitting each word in separate audio file – .WAV file) has been a major concern when working with multimedia such as recordings from radio or TV. The main focus of the segmentation of boundaries of spoken language has been on using energy and zero crossing thresholds for endpoint detection. Errors in endpoint detection are still a main cause of low accuracy of segmentation systems. The goal of this research is to develop an efficient algorithm in order to segment the speech of human in both languages of English and Arabic in different speaking speed with high accuracy. Simulation results show that the developed algorithm achieved high accuracy when segmenting human speech in English language up to 91.6% in average, while it is 89.0% of Arabic language.

Keywords: Audio, Voice, Speech, Segmentation

Introduction

Speech is a primary means of communication between humans. In the information society, speech is used not only in its original form but also through many digital electronic devices (Zhang and Kuo, 2001) mobile phones grow rapidly, wired telephone systems become digital and Internet phones and audio are common in use. On the other hand, computers may also speak to humans by synthetic voice (Juan *et al*., 2015) and listen to us using speech recognition. To understand these processes for both human and machine, we have to study carefully the structures and functions of spoken language: how to produce and perceive it and how speech technology may help us to communicate (Hennig and Chellali, 2012)

Speech segmentation is the process of splitting the speech into separately words and each word is saved in separated audio file for the upcoming processing as shown in Fig. 1. Speech segmentation becomes an interesting area of research. Whereas several algorithms work on audio files, mainly in English language, which achieve different accuracy of segmentation based on the properties of the spoken languages themselves.

In this research, we developed a fast and simple-to-implement segmentation algorithm that matches closely subjective expectations of the required target. The algorithm is based on two thresholds that are benefit from a dynamic window in order to split the words correctly. The algorithm is implemented using MatLab and is applied on Arabic and English Languages, the accuracy of algorithm is high.

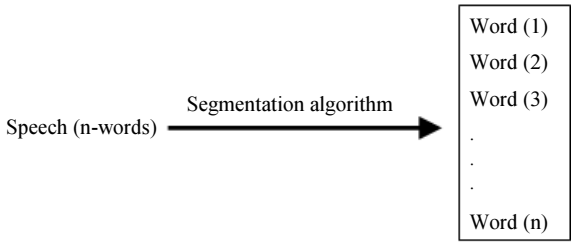

Fig. 1: The output of segmentation

The remaining of the paper is described as follows: part II provides an overview about audio signal and sampling. Part III describes the related work. Part IV describes the proposed algorithm. Part V describes the experiments and the evaluation of results. Part VI shows the complexity of algorithm and part VII is the conclusion.

Related Work

Several researches and methods have been done to improve the performance of speech segmentation (Naoki *et al*., 2006) proposes an audio signal segmentation and classification method using fuzzy c-means clustering. (Shi-sian and Hsin-min, 2003) proposes a sequential metric-based audio segmentation method that has the advantage of low computation cost of metric-based methods and the advantage of high accuracy of model-selection-based methods.

Other approaches are based on Hidden Markov Model (HMM) (Daniel and James, 2017) such as (Lefevre *et al*., 2002) that combines a K-Means classifier with Hidden Markov Models in order to analyze audio segment using several audio features based either on segment or frame. Another method base on HMM is (Biswajit *et al*., 2015) that aims at exploring Vowel Onset Point (VOP) and Vowel offset or End Point (VEP) for correcting the boundaries obtained using HMM alignment. HMM models the class information well, but it may not detect the exact boundary. Another method based on HMM and designed for Arabic language is (Abed *et al*., 2016) that proposes an automatic segmentation system of speech into phonemes for the Arabic language. This segmentation is based on two different techniques: Hidden Markov Models (HMM) and Artificial Neural Networks (ANN). Both systems were used to classify the speech signals, extracted from ALGerian Arabic Speech Database (ALGASD corpus), into five classes: Fricatives, plosives, nasals, liquids and vowels.

Adriana *et al*. (2015) introduces a first attempt to perform phoneme-level segmentation of speech based on a perceptual representation - the Spectro Temporal Excitation Pattern (STEP) - and a dimensionality reduction technique - the t-distributed Stochastic Neighbor Embedding (t-SNE). The method searches for the true phonetic boundaries in the vicinity of those produced by an HMM-based segmentation. It looks for perceptually-salient spectral changes which occur at these phonetic transitions and exploits t-SNE's ability to capture both local and global structure of the data.

Some methods benefit from sliding window technique such as (Shih-sian and Hsin-min, 2004), which presents a hybrid approach for audio segmentation, in which the metric-based segmentation with long sliding windows is applied first to segment an audio stream into shorter sub-segments and then the divide-and-conquer segmentation is applied to a fixed-length window that slides from the beginning to the end of each sub-segment to sequentially detect the remaining acoustic changes. (Bartosz *et al*., 2006) applies the Discrete Wavelet Transform (DWT) to analyze speech signals, the resulting power spectrum and its derivatives. This information allows locating the boundaries of phonemes.

Ghazaal and Farshad (2011) investigates the problem of segmenting speech into sub_word units. a technique based on fuzzy smoothing is applied on short term energy function of speech wave. The smoothed energy contour is searched then in order to find local minima that imply to syllable units.

Benati and Halima (2016) suggests the use of an acoustic-based algorithm for the segmentation which exploits acoustic particularities of the speech stream to detect word frontiers.

Fréjus *et al*. (2015) presents an algorithm using fuzzy logic approach to perform the continuous speech segmentation task from non-linear speech analysis. The proposed algorithm is based on time domain features. These features are the short-term energy, zero crossing rate and the singularity exponents calculated in each point of signal. While (Brognaux and Thomas, 2016) focuses on a particular case of hidden Markov model (HMM)-based forced alignment in which the models are directly trained on the corpus to align.

Kamper *et al*. (2017) introduces an approximation to a recent Bayesian model that still has a clear objective function but improves efficiency by using hard clustering and segmentation rather than full Bayesian inference.

Kiss *et al*. (2013) introduces a language independent solution that based on the segmentation of continuous speech into 9 broad phonetic classes. The classification and segmentation was prepared using Hidden Markov Models.

Proposed Method

The algorithm aims to split the audio file that contains several words into several audio files that each one contains one word only.

The process of segmentation goes through several steps; starting by converting the audio signal into samples and then performing normalization process (Lucero and Koenig, 2000) on them to handle the irregularity of samples amplitude, then check for muteness periods, next check for the continuous words as the last step before saving the segmented samples into separate file as shown in Fig. 2.

Figure 3 illustrates the flowchart that is explained in more details.

The first step after sampling is the normalization; Equation 1 converts samples into their normalization values:

$$f_n = f_s * H / f_h \quad (1)$$

Where:

F_n = The amplitude of normalized sample
F_s = The amplitude of sample before normalization
H = Normalization level
F_h = The highest sample's amplitude

The algorithm then applies two thresholds and the dynamic window:

The algorithm depends on two threshold values; the first one segments the slow speech, which has a considerable period of muteness, while the second one segments the speech that is relatively fast.

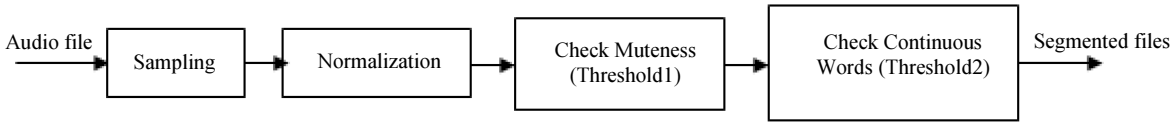

Fig. 2: Main processes of the algorithm

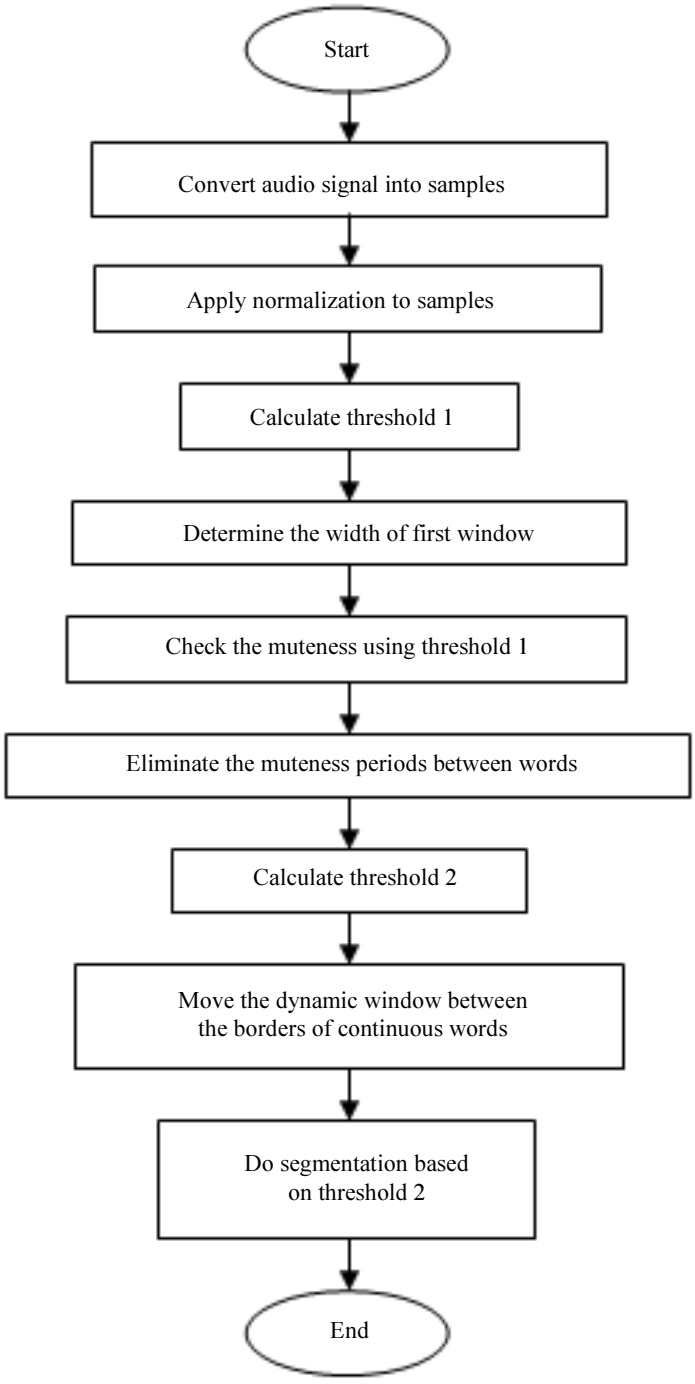

Fig. 3: Flowchart of the proposed algorithm

The first threshold is determined based on the average value of samples' amplitude as in Equation 2:

$$Th1 = 0.55 * \frac{\sum_{i=0}^{j} f_n}{j} \quad (2)$$

Where:
$Th1$ = The value of threshold 1
f_n = The amplitude of the normalized sample
j = The total number of samples

The percentage (0.55) is measured based on many tests; and after many experiments trying several percentages, the 55% achieves the best performance in both languages.

Next step is to determine the width of first window; the initial width of window is equal to the average length of spoken word (the width is varying based on the spoken language); and then starting from the beginning of samples and testing the amplitude at the end of window, the windows' width is extended until measuring muteness.

The first threshold is used to check the muteness occurrences; this is done by moving the window through the samples and then obtaining the maximum value and comparing it to the threshold. This technique is used to make sure that the highest sample value is selected each time. If that value is greater than the threshold, then it is a continuous word and the algorithm continues checking next values by moving the window to next position until a value less than the threshold is obtained, which means end of word(s) as described below:

1. Set the starting point of samples.
2. Compare (H) the highest value of samples within the window to (Th1) threshold1
   If W is less than Th1
      Delete the muteness
      Split the words
   Else
      Move the window and go to step 2.
3. Repeat until the end of samples

Next step is to eliminate the muteness periods between words; which has an average time of 1000 samples (this number is obtained by testing many audio wave files in Arabic and English languages) and checking for that number of samples to eliminate these unusable segments as shown below:

1. If H is less than Th1
   Add 1000 to W
2. Delete samples between W and W+1000
3. Move the window to W+1000 position
4. Go to step 1

The algorithm then analyzes the muteness between long segments in order to adjust the width of window; the shortest word in spoken Language (Arabic and English languages) is measured through the whole audio file by determining the number of reasonable amplitudes between two continuous muteness intervals as shown below:

1. Set M the end of muteness
2. Set N the start of next muteness
3. calculate the width of window = N-M

A second threshold is used to segment the previous continuous words as shown in Fig. 3, the value of second threshold is the average amplitude of samples (Equation 3):

$$Th2 = \frac{\sum_{i=0}^{j} f_n}{j} \quad (3)$$

The same idea is used by moving the dynamic window between the borders of continuous words –i.e., the start and end indexes- as shown in Fig. 4 and determining the next value which should be less than the second threshold, which means that it is a complete word and segmentation should be done here.

The previous steps are repeated to the end of samples in order to segment all the words as shown below:

1. Set the starting point of muteness M
2. Set the end point of muteness K
3. Move window
4. Compare the highest sample value (H) to threshold 2 (Th2)
   If H<Th2
      Split the word and save it to .wav file
   Else
5. Go to step 3

Experiments and Evaluation

The proposed algorithm is applied using MatLab on 50 .wav audio files of English language and another 50 .wav audio files of Arabic language, the speed of speech changes from fast to slow in both languages. Table 1 shows a sample of tested files of English language and the accuracy of segmentation.

The accuracy of segmentation for English language is 91.6%.

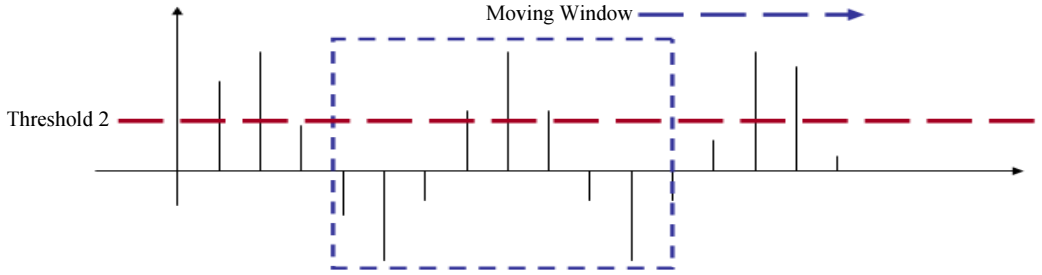

Fig. 4: Dynamic moving window with threshold2

Table 1: Sample of experimental results – English Language

No.	Period (seconds)	No. of Words/File	No. of words segmented correctly
1	9	12	11
2	8	5	5
3	6	5	3
4	6	8	8
5	3	7	6
Total no. of words		37	33

Table 2: Sample of experimental results – Arabic Language

No.	Period (seconds)	No. of Words/File	No. of words segmented correctly
1	10	13	11
2	3	4	3
3	5	5	4
4	4	6	6
5	2	7	6
Total no. of words		35	29

Table 2 shows sample of tested files of Arabic language and the accuracy of segmentation is 89.0%.

The accuracy of segmentation of Arabic files are not better than English ones due to the nature of language itself; Arabic language has vowels, which take relatively longer time than its counterpart of English, the algorithm may consider them as muteness and so the algorithm can fail in determining the case of muteness.

These results and accuracy can vary depending on the speed of speech and the spoken language itself.

Algorithm Complexity

This section aims to measure the performance of algorithm using the Big *O* Notation []. Based on the analysis of algorithm, it belongs to the $O(n2)$; the algorithm has some single loops and two nested loops and so the effective part is the nested loops and so its complexity is $O(n2)$.

Conclusion

In this research, a new segmentation technique is introduced that depends on two threshold values that are calculated based on the samples themselves and two windows of width that are measured based on the muteness of audio samples.

In this technique, we aim to go through the peak values of samples by means of dynamic moving window that capture number of samples and calculate the maximum amplitude within the window and so we guarantee that the algorithm does not fail with samples less than threshold (part of word).

The first threshold determines the muteness when a suitable period of muteness is exist and then the algorithm splits words when that period is too small, the idea used is to go through another threshold higher than the first one, in addition to a window size less than the previous one to have the capability to capture the small period of time between two words.

The algorithm is tested on Arabic and English languages whereas the accuracy of English language is better than Arabic one due to the nature of language itself.

Acknowledgment

This research was supported and funded by Arab Open University - Kuwait Branch. We thank everybody who assisted us to improve the study.

Ethics

I testify that my research paper submitted to the Journal of Science Publication, title: Speech

Segmentation Using Dynamic Windows and Thresholds for Arabic and English Languages has not been published in whole or in part elsewhere.

This research project was conducted with full compliance of research ethics norms of Arab Open University - Kuwait.

References

Abed, A., A. Amrouche, A. Delmadji, K. Boubakeur and G. Droua-Hamdani, 2016. Automatic segmentation of Arabic speech signals by HMM and ANN. Proceedings of the International Conference on Electrical Sciences and Technologies in Maghreb, Oct. 26-28, IEEE Xplore Press, Marrakech, Morocco, pp: 1-4. DOI: 10.1109/CISTEM.2016.8066776

Adriana, S., V.B. Cassia, G. Mircea and K. Simon, 2015. Phonetic segmentation of speech using STEP and t-SNE. Proceedings of the International Conference on Speech Technology and Human-Computer Dialogue, Oct. 14-17, IEEE Xplore Press, Bucharest, Romania, pp: 1-6. DOI: 10.1109/SPED.2015.7343105

Bartosz, Z., M. Suresh, C.W. Richard and Z. Mariusz, 2006. Wavelet method of speech segmentation. Proceedings of the 14th European Signal Processing Conference, Sept. 4-8, IEEE Xplore Press, Florence, Italy, pp: 1-5.

Benati, N. and B. Halima, 2016. Spoken term detection based on acoustic speech segmentation. Proceedings of the 7th International Conference on, Sciences of Electronics, Technologies of Information and Telecommunications, Dec. 18-20, IEEE Xplore Press, Hammamet, Tunisia, pp: 267-271. DOI: 10.1109/SETIT.2016.7939878

Biswajit, D.S., S.S. Bidisha, S. Aswin, S.R. Mahadeva Prasanna and A.H. Murthy, 2015. Exploration of vowel onset and offset points for hybrid speech segmentation. Proceedings of the IEEE Region 10th Conference TENCON, Nov. 1-4, IEEE Xplore Press, Macao, China, pp: 1-6. DOI: 10.1109/TENCON.2015.7373137

Brognaux, S. and D. Thomas, 2016. HMM-based speech segmentation: Improvements of fully automatic approaches. I. ACM Transact. Audio, Speech Langu. Proces., 24: 5-15. DOI: 10.1109/TASLP.2015.2456421

Daniel, J. and H.M. James, 2017. Hidden Markov models. Stanford University.

Fréjus, A.A.L., C.E. Eugène and M. Cina, 2015. An algorithm based on fuzzy logic for text-independent fongbe speech segmentation. Proceedings of the International Conference on Signal-Image Technology and Internet-Based Systems, Nov. 23-27, IEEE Xplore Press, Bangkok, Thailand, pp: 1-6. DOI: 10.1109/SITIS.2015.72

Ghazaal, S. And A. Farshad, 2011. Segmentation of speech into syllable units using fuzzy smoothed short term energy contour. Proceedings of the 18th Iranian Conference of Biomedical Engineering, Dec. 14-16, IEEE Xplore Press, Tehran, Iran, pp: 195-198. DOI: 10.1109/ICBME.2011.6168554

Hennig, S. and R. Chellali, 2012. Expressive synthetic voices: considerations for human robot interaction. Proceedings of the 21th International Symposium on Robot and Human Interactive Communication, Sept. 9-13, IEEE Xplore Press, Paris, France, pp: 589-595. DOI: 10.1109/ROMAN.2012.6343815

Juan, G., P. Nicola, S. Lucas and H. Santiago, 2015. Synthetic voice harmonization: A fast and precise method. Proceeding of the International Symposium on Multimedia, Dec. 14-16, IEEE Xplore Press, Miami, FL, USA, pp: 81-84 DOI: 10.1109/ISM.2015.122

Kamper, H., L. Karen and G. Sharon, 2017. An embedded segmental k-means model for unsupervised segmentation and clustering of speech. Comput. Sci., Comput. Lang.

Kiss, G., S. Dávid and V. Klara, 2013. Language independent automatic speech segmentation into phoneme-like units on the base of acoustic distinctive features. Proceedings of the 4th International Conference on, Cognitive Infocommunications, Dec. 2-5, IEEE Xplore Press, Budapest, Hungary, pp: 579-582. DOI: 10.1109/CogInfoCom.2013.6719169

Lefevre, S., B. Maillard and N. Vincent, 2002. A two level classifier process for audio segmentation. Proceedings of the 16th International Conference on Pattern Recognition, Aug. 11-15, IEEE Xplore Press, Quebec City, Canada, pp: 891-894. DOI: 10.1109/ICPR.2002.1048175

Lucero, J.C. and L.L. Koenig, 2000. Time normalization of voice signals using functional data analysis. J. Acoustic. Society Am., 108: 1408-20. PMID: 11051467

Naoki, N., H. Miki and K. Hideo, 2016. Audio signal segmentation and classification using fuzzy c-means clustering. J. Syst. Comput. Japan, 37: 23-34. DOI: 10.1002/scj.20491

Shih-sian, C. and W. Hsin-min, 2004. Metric-seqdac: A hybrid approach for audio segmentation. INTERSPEECH.

Shi-sian, C. and W. Hsin-min, 2003. A sequential metric-based audio segmentation method via the Bayesian information criterion. Proceedings of European Conference of Speech Communication and Technology, (SCT' 03).

Zhang, T. and C.C.J. Kuo, 2001. Audio content analysis for online audiovisual data segmentation and classification. I. Transacti. Speech Audio Proces., 9: 441-457. DOI: 10.1109/89.917689