Audio Segmentation with Bayesian Inference Criterion

Pankhuri Vanjani, Navami Kairanda

Universität Des Saarlandes

1 Introduction and Problem statement

A variety of audio processing tasks require the audio to be segmented into segments which have acoustically similar properties. Particularly in the speaker identification, environmental change detection, speech clustering it is important to automatically detect the points of acoustic change and segment them appropriately before the further processing. These tasks have applications in audio transcriptions, music classification or speech monitoring for surveillance purpose.

Environmental factors like noise make the segmentation task more challenging, besides the sounds which are for smaller duration in the audio stream are often not segmented properly. This work focuses on speech segmentation of an audio clip while looking into these challenges too. It follows the work of the [1] and [5] using Bayesian Inference Criterion.

2 Background Information

Features are an important components in for audio classification and analysis, they help in finding relations and analysis, they are broadly classified in 2 type physical and perceptual features [6]. Physical features are mathematically derived from the sound, some of them are cepstral coefficients, spectrum, energy functions. Perceptual features are the way humans perceive the sound as in loudness, timbre, pitch etc. Some of the relevant features are discussed below:

1.Mel frequency cepstral coefficients (MFCCs) Cepstrum gives the information about change in rate of spectral bands and Mel scale plays the role of relating a tone's perceived frequency with the actual measured frequency. MFCCs contains set of coefficients to describe the Mel Frequency cepstrum. Essentially, they are set of features describing the spectral envelope shape

2. Spectral Centroids Spectral centroids indicate the center of mass for audio signals, it is calculated by taking the weighted average of frequencies. For a signal with frequency spread uniformly center of mass lies in the middle and of other cases it lies closer to the high-frequency signals.

$$\text{centroid } = \left(\sum_{k=b_1}^{b_2} f_k s_k\right) / \left(\sum_{k=b_1}^{b_2} s_k\right)$$

f_k: frequency at bin k, s_k : spectral value at bin k, $b1, b2$: band edges

3. Spectral Rolloffs Spectral rolloffs indicates the frequency below which N (usually N= 85 or 95)percentage of the total spectral energy of the signal lies.

$$\text{rolloff } = \sum_{k=b_1}^{d} s_k = N \left(\sum_{k=b_1}^{b_2} s_k \right)$$

N: Percentile cutoff s_k : Spectral value at bin k b1, b2 : Band edges

4. RMS Features Root Mean Square features correspond to the energy of the signal or total magnitude in mathematical terms. Here,

$$\text{RMS}_t = \sqrt{\frac{1}{N} \cdot \sum_{k=t \cdot N}^{(t+1)\cdot(N-1)} s(k)^2}$$

For segmentation tasks, researchers have often used classifiers but they don't perform well on unseen data. Bayesian Inference Criterion (BIC) proposed by [1] works well in handling such unseen data as well as improves accuracy and speed.

Bayesian Inference Criterion:

BIC is a model Selection criterion to select appropriate model which can describe the set of data well. There are different models available for a task, some of them give good results but they have huge number of parameters. Increased parameters often lead to the problem of over training, so it is important select a model which balances both complexity as well as accuracy. One such parametric method of selection of models is Bayesian Inference Criterion which has been used in this work. It is based on maximum likelihood and penalizes the complexity in a model (which is related by number of model parameters). BIC can be mathematically formulated as described below:

$$BIC(M) = \log L(\mathcal{X}, M) - \lambda \frac{1}{2} \#(M) \times \log(N)$$

where, $\mathcal{X} = \{x_i : i = 1, \cdots, N\}$ the dataset being modelled

$\mathcal{M} = \{M_i : i = 1, \cdots, K\}$ model choices

$L(\mathcal{X}, M)$ Likelihood function of each model

λ Penalty weight and $\#(M)$ Number of parameters for a model M

BIC thresholds the log-likelihood distance, where level for thresholding is automatically selected with the formula given below. Here d= Feature space dimension, N= window size.

$$\lambda_* 0.5 \left(d + 0.5 * d(d+1)\right) \log N$$

3 Methodology and Experimental Results

A higher level view of Audio segmentation performed in this work has been demonstrated in the figure 1. From the audio file, relevant features are extracted

first which are passed into BIC computation module. An important part of this system is that features are passed by dividing into some window size. Tuning the window size rightly plays an important role in getting accurate segmentation results. After getting BIC the BIC value decision is made for changing the starting-ending points of next window, also number of frames by which window size is growing is changed in this decision module. If it is able to detect the changing segment boundary, the next window starts after the detected boundary points, otherwise variable window increasing scheme is used. These detected boundary points give the segmented audio clips as well as indicate on the spectrum.

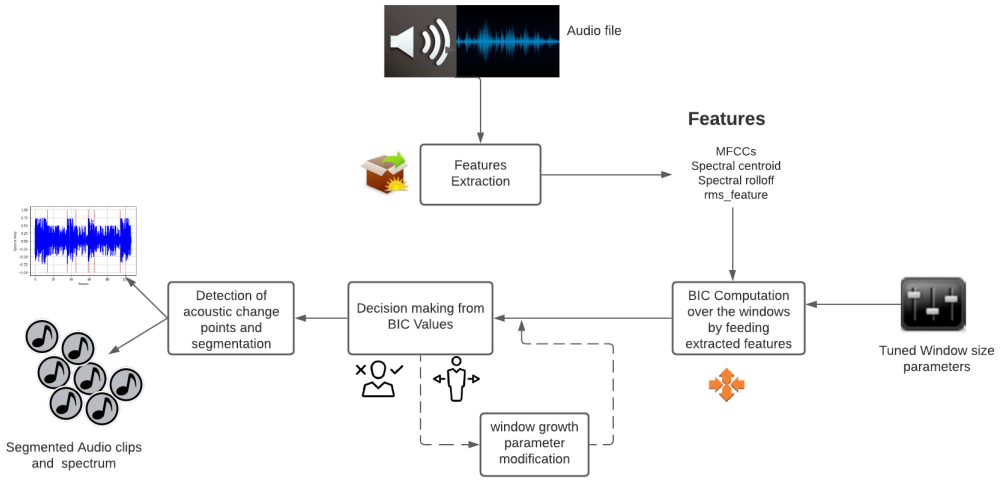

Fig. 1. Methodology diagram

3.1 Feature Extraction

For feature Extraction, python based librosa library has been used. The sampling rate of audio clip comes out to be 48,000Hz.

Important parameters calculation Frame width of 25ms and frame shift of 10ms has been provided, when converting them to data samples, we get:

Frame width: width*sampling rate= $0.025 * 48000 = 1200$ Taking the data samples as power of 2, for 1200 we take the upper bound, framewidth as 2048. Frame shift: shift*sampling rate $= 0.010 * 48000 = 480$ Taking the data samples as power of 2, for 480 we get the upper bound frameshift as 512.

Feature Selection and concatenation MFCCs are the most important features for the segmentation tasks, they are sufficient to get results but adding the other features enhances the feature bank and performance. In this module besides MFCCs, Spectral centroid, spectral rolloff and RMS Energy features have been used, experimentally they have given the best results combined together. [2] has provided a good overview of audio features which has helped in selection.

- RMS Energy Features: It's loudness perceiving and robustness against outliers characteristics make it useful for detecting changes in multi person sound file reliably.
- Spectral Centroid: It's robustness with the brightness of sound results from [4] made it a good candidate for experiments in this work.
- Spectral rolloff: This feature was introduced in [3] and has been shown to give good performance in acoustic segmentation along with MFCCs.

The size of MFCCs is (24 , 9963), spectral centroid and rolloff is (9963 ,) and RMS is (1 , 9963). Features have been augmented by increasing the dimensions along MFCCs making the final feature dimensions as (27 , 9963)

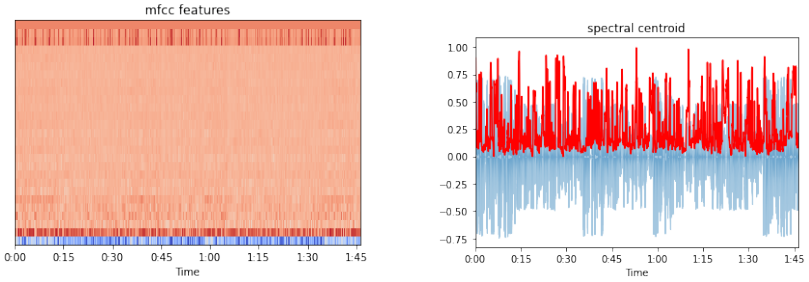

Fig. 2. MFCC(Left)and Spectral centroid(Right) plot

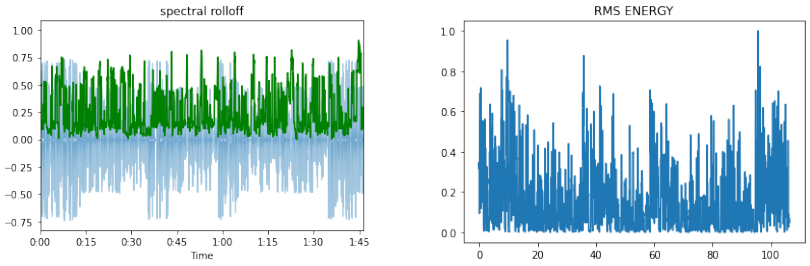

Fig. 3. Spectral centroid(Left)and RMS (Right) plot

3.2 BIC Computation

For BIC computation following formula has been used in this work which is based on [5]. Following parameters effected the results, and through several experiments results were improved.

Effect of Penalty weight Penalty weight denotes by λ in the above equation. Initially lambda value was taken as 1. It was gradually increased as 1, 1.3, 1.5 and 1.6 gave the best results. Values lesser than 1.6 gave a lesser segment.

- = 1/1.3/1.5 = 100: 5 correct detection, 0 false detection, 2 undetected
- **=1.6/1.7**: 6 correct detection, 0 false detection, 1 undetected

Window Size Tuning As the paper suggested, initially we started with window size = 100 and increased it to optimal results which were obtained for **window size = 200**.

- window size = 100: 6 correct detection, 1 false detection (around 64 seconds), 1 undetected
- window size = 150: 6 correct detection, 1 false detection (around 2 seconds)

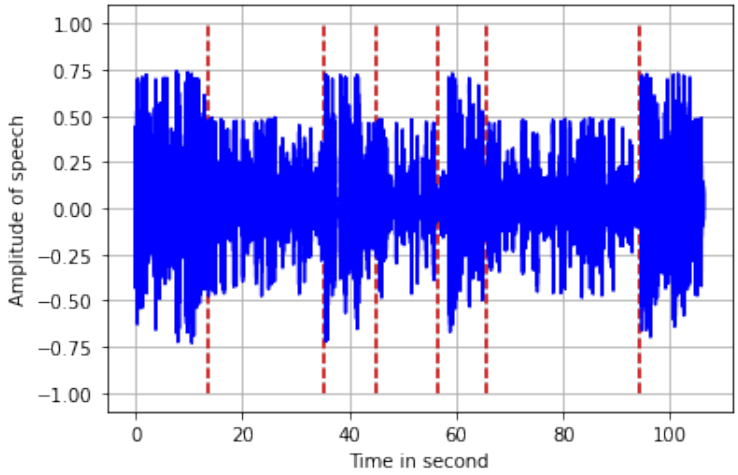

Fig. 4. Best result with parameters

Manually Verified Detection Points (in seconds)(approximately)	13	35	44	57	64	95	104	ending
Automatic Detection points from the (in seconds)	13.610	35.381	44.992	56.469	65.546	94.336	N/D	ending

Table 1. Comparison of manually detected points vs automatically detected point for acoustic changes in audio segmentation for 107 seconds audio file.

Efficient BIC Tests As the paper [5] suggested efficient BIC tests to reduce the redundant tests, in the BIC Computation a small code snippet has been added which tests if the n is greater than 2 times the threshold parameter(which means if the window is large) then instead of going over all the samples in the large window, computation is done from middle(i.e. we don't compute the half of the initial elements). Thresholding parameter has been taken as 100 here and this gives the same accuracy but improves the computation speed by almost 2 times as reported below:

- For vanilla BIC: 0.201 seconds
- Efficient Tests BIC: **0.11 seconds**

3.3 Variable window growing method in decision making for Segmentation

Initially, number of frame by which window was growing (ΔN_i) was kept fixed as 100 but using the variable window scheme, it has been changed by $\Delta N_i = \Delta N_{i+1} + \delta_i$ where δ_i is updated as $\delta_i = 2\delta_{i+1}$. Without this scheme, with parameter tuning decent results were obtained but there was improvement in computation speed too which can be seen below. $\delta_i = \mathbf{12}$ has given the best results.

- For fixed growing window size: 0.335 seconds
- For variable window growing: **0.11** seconds

With the best parameters determined from the above experiments Fig.4 and Table1 show the final best results obtained in this work. Ideally there are 7 detection points and 8 segments, wheras we are getting 6 detection points and 7 segments.

4 Conclusion

In the following work, Audio segmentation was performed using Bayesian Inference Criterion for an audio clip which had multiple speakers. Through the experiments it was concluded feature selection is an important criteria in audio-preprocessing and for the BIC based segmentation part the performance was improved by tuning parameters like penalty weight, adjusting window sizes. Also,

we were able to reduce the computation time using variable window growing method and efficient BIC tests. The 107 seconds audio segmentation clip used for testing had 8 segments when tested manually, this work has been able to get 7 segments where the last 2 seconds are being merged with the segment immediately before. The rest of the 7 segments have been very accurately detected, besides there is no false detection which shows good performance of the algorithm. One of the reasoning behind the misspoint is that the last segment is of very small duration(approximately 2-3 seconds), in the reference work [5] also authors were getting 13.1% of misspoints for less than 2 seconds of segments. With current deep learning based models, there is a future scope of this work of improving this accuracy further.

References

1. Chen, S., Gopalakrishnan, P., et al.: Speaker, environment and channel change detection and clustering via the bayesian information criterion. In: Proc. DARPA broadcast news transcription and understanding workshop. vol. 8, pp. 127–132. Virginia, USA (1998)
2. Giannakopoulos, T., Pikrakis, A.: Introduction to audio analysis: a MATLAB® approach. Academic Press (2014)
3. Kos, M., Kačič, Z., Vlaj, D.: Acoustic classification and segmentation using modified spectral roll-off and variance-based features. Digital Signal Processing **23**(2), 659–674 (2013)
4. Le, P.N., Ambikairajah, E., Epps, J., Sethu, V., Choi, E.H.: Investigation of spectral centroid features for cognitive load classification. Speech Communication **53**(4), 540–551 (2011)
5. Tritschler, A., Gopinath, R.A.: Improved speaker segmentation and segments clustering using the bayesian information criterion. In: Sixth European Conference on Speech Communication and Technology (1999)
6. Zhang, T., Kuo, C.C.J.: Audio feature analysis. In: Content-Based Audio Classification and Retrieval for Audiovisual Data Parsing, pp. 35–54. Springer (2001)