NOVEL STRATEGIES FOR REDUCING THE FALSE ALARM RATE IN A SPEAKER SEGMENTATION SYSTEM

Paula Lopez-Otero, Laura Docio-Fernandez, Carmen Garcia-Mateo

E.T.S.I. Telecomunicacion
Department of Signal Theory and Communications, University of Vigo
Campus Universitario de Vigo, 36310 Vigo, Spain

ABSTRACT
Reliable speaker segmentation is critical in many applications in the speech processing domain. In this paper, we extend our earlier formulation for false alarm reduction in a typical state-of-art speaker segmentation system. Specifically, we present two novel strategies for reducing the false alarm rate with a minimal impact on the true speaker change detection rate. One of the new strategies rejects, given a discard probability, those changes that are suspicious of being false alarms because of their low $\Delta BIC$ value; and the other one assumes that the occurrence of changes constitute a Poisson process, so changes will be discarded with a probability that follows a Poisson cumulative density function. Our experiments show the improvements obtained with each false alarm reduction approach using the Spanish Parliament Sessions defined for the 2006 TC-STAR Automatic Speech Recognition evaluation campaign.

Index Terms— Audio segmentation, speaker segmentation, speaker change detection, speaker diarization.

1. INTRODUCTION
Nowadays, an emerging application area where speech technologies are involved is the field of structuring the information of multimedia (audio-visual) documents. These multimedia documents are, in general, multi-speaker audio recordings, and for some applications it may be relevant to determine “who spoke when”. This task is also referred to as “speaker segmentation and clustering” or “speaker diarization” in the literature. The segmentation of the data in terms of speakers could help in efficient navigation through audio documents, such as meeting recordings or broadcast news archives. Using these segmentation queues, an interested user can directly access a particular segment of the speech spoken by a particular speaker. Other applications of the speaker segmentation task include speaker adaptation in speech recognition and speaker identification-verification-tracking.

In this paper, we focus on a common problem that appears in state-of-art speaker segmentation systems: the false alarms. The appearance of false alarms causes an over-segmentation of data, which will harm the effectiveness of the task that will use the segmented data. For example, false alarms can reduce the accuracy of a speech recognizer, as the over-segmentation of data can cause the language models not to work appropriately. As well, in a speaker identification-verification-tracking task, the appearance of short segments caused by false alarms can guide to a poor speaker modeling due to the lack of enough data.

In [1] we introduced an online four-stage speaker segmentation system that first performs a coarse segmentation of the data, then refines or discards the change points, discriminates between speech and non-speech, and merges segments that are likely to be spoken by the same speaker. We noticed that this baseline segmentation system has a high false alarm rate and tends to estimate short segments. In this work, to avoid erroneous speaker changes we introduce and evaluate two novel approaches for reducing the number of false alarms in our speaker segmentation system, and compare them with the false-alarm discard algorithm proposed in [1]. The first approach rejects, given a discard probability, those changes that are suspicious of being false alarms because of their low $\Delta BIC$ value. As well, the second strategy assumes that the occurrence of changes constitute a Poisson process, so changes will be discarded with a probability that follows a Poisson cumulative density function. The goal of such techniques is to confirm true speaker changes and suppress erroneous speaker changes.

The paper is organized as follows. Section 2 gives a brief description of the baseline speaker segmentation system. The proposed approaches to reduce the false alarm rate are presented in Section 3. In Section 4 we give an explanation of our experimental framework. The performance of the speaker segmentation system using each one of the false alarm reduction strategies are shown and discussed in Section 5. Finally, Section 6 concludes this paper and gives some ideas of future work.

2. ARCHITECTURE OF THE SPEAKER SEGMENTATION SYSTEM
The architecture of the baseline speaker segmentation system described in [1] is depicted in Fig. 1, where we can see it has four phases: first, a coarse segmentation is made with the Distance Changing Trend Segmentation algorithm (DCTS) [2] in order to detect audio change-point candidates and then a refinement or rejection of these change-point candidates is performed by the Bayesian Information Criterion (BIC) algorithm [3]. After that, the system makes a Maximum a Posteriori (MAP) adaptation of three different Gaussian Mixture Models (GMMs) to decide whether the audio segment delimited by the new change-point and the preceding one is speech, music or silence/noise. If the segment is speech, the same procedure will be employed to classify the speech in male or female speech. Finally, when the two latest segments are speech an approach based on the Cross Likelihood Ratio (CLR) [4] test is applied in order to find out if both speech segments are spoken by the same speaker; in that case both speech segments are merged.

This paper has been partially supported by Spanish MCI under the project TEC2006-13694-C03-03/TCM.
3. FALSE ALARM REJECTION STRATEGIES

The proposed strategies to discard false alarms are based on the monitoring of the $\Delta BIC$ value

$$\Delta BIC(i) = L(i) - \lambda P \quad (1)$$

where $P$ is the penalty, corresponding to the number of free parameters of the Gaussian model, and $\lambda$ is a weight that increases or decreases the influence of the penalty. When $\lambda$ is a small value, less changes will be discarded by the BIC algorithm; the opposite happens when $\lambda$ gets bigger.

Equation (1) has a member $L(i)$ which represents a likelihood ratio:

$$L(i) = \frac{L_2}{2} \log |\Sigma_2| - \frac{L_1}{2} \log |\Sigma_1| - \frac{L_2}{2} \log |\Sigma_2| \quad (2)$$

where $L$, $L_1$ and $L_2$ are the frame sizes of segments $X$, $X_1$ and $X_2$ respectively; and $\Sigma$, $\Sigma_1$ and $\Sigma_2$ are the covariance matrices for the models $M$, $M_1$ and $M_2$ respectively. Thus, there will be a change in the audio data when

$$\frac{L_2}{2} \log |\Sigma_2| - \frac{L_1}{2} \log |\Sigma_1| - \frac{L_2}{2} \log |\Sigma_2| > \lambda P \quad (3)$$

3.1. Adaptive threshold-based rejection

The goal of this approach proposed in [1] is to increase change rejections when the number of changes discarded by the CLR test increases, and reduce them when this value remains the same. It is equivalent to modify dynamically the value of the parameter $\lambda$. To achieve this, four new parameters are introduced: a discard probability $p_{\text{discard}}$, a threshold $T$, a counter for the removed points $E$, and a counter for the non-removed points $N$. The algorithm works as follows:

1. Initially, $p_{\text{discard}} = 0$ and $T = 0$, that means, no change will be discarded.
2. When DCTS detects a change, BIC algorithm will be run to refine or discard the change. If it is not discarded, the value returned by BIC will be stored.
3. When we have two consecutive speech segments, we will check if they belong to the same speaker, so as to merge them in one segment. We may have two different situations:

(a) If the CLR computed is above a threshold, it will mean that the two segments are similar so they will be merged, that is, the change between them will be discarded. When that happens we update $T$ and $p_{\text{discard}}$:

$$E = E + 1 \quad (4)$$

$$p_{\text{discard}} = \frac{E}{N + E} \quad (5)$$

$$T = \sum_{i} \text{threshold}_{E} \quad (6)$$

where $\text{threshold}_{E}$ is the value returned by BIC for the $E^{th}$ segment. Equation (5) will make $p_{\text{discard}}$ increase, so a change will be discarded more easily.

(b) If the CLR is below the threshold, the segments are supposed to belong to different speakers, so we keep the change between them and recompute $p_{\text{discard}}$:

$$N = N + 1 \quad (7)$$

$$p_{\text{discard}} = \frac{E}{N + E} \quad (8)$$

Equation (8) will assign a lower value to $p_{\text{discard}}$, so it will be more difficult to discard a change.

4. Anytime the BIC algorithm detects a change point, we will decide whether to discard it or not:

$$BIC \rightarrow V \quad \text{(decision value)}$$

If $V < T$:

(a) Compute a random number $r$.

(b) If $r < p_{\text{discard}}$, the change will be discarded.

This algorithm had two problems that could be solved easily. It was observed that when the BIC algorithm returned value $V$ was very low, this change usually was an insertion. So, any change with $V < 100$ will be automatically discarded. Also, we found the need to limit the upper value of $T$, because it would get greater and greater and discard changes that are not insertions. After some experiments on the training set, we found that the best value for the upper threshold of $T$ was 250.

3.2. Uniform distribution-based rejection

In this change-rejection approach, we will suppose that false alarms follow a uniform distribution, i.e. all changes with their $\Delta BIC$ value below a given threshold have the same probability of being a false alarm. To suppress false alarms this approach works as follows: for each change-point to be tested, first a uniformly distributed pseudo-random number is generated; and then that change is discarded if that number is below a fixed threshold. That threshold controls the probability assigned to the hypothesis “the change is a false alarm”. So, if it is 0.5 the two alternatives “the change is a false alarm” and “the change is not a false alarm” are equally probable. When the threshold increases to, for example, 0.75, the probability of the alternative “the change is a false alarm” is increased. This fact reduce the number of false alarms but also increases the number of deletions, as a real change will be easily rejected. When we choose a lower threshold, for example 0.2, the number of false alarms will increase, but the number of miss detections will be lower. Then, we can select how tolerant we want to be with false alarms: when we choose a high threshold, we will reject a high percentage of false
alarms, and with a low threshold we will tolerate a high number of false alarms, being more strict with deletions.

We can see that this is the same as done in point 4 of the preceding system, but with the difference that here there will be no modifications on the discard probability nor on the threshold.

### 3.3. Poisson distribution-based rejection

In this false alarm suppression strategy we assume that the occurrence times of change-points can be modelled by a Poisson process. A Poisson process is an independent occurrence process where the number of occurrences in two disjoint time intervals is independent, the probability of having an occurrence is proportional to the observed interval, and the occurrences are not simultaneous.

The process we are dealing with in speaker segmentation fulfills those four properties, as it is a process where arrivals (of changes) happen independently from the others and in random instants. Poisson processes have a probability density function

\[ f(\mu, x) = \frac{e^{-\mu} \mu^x}{x!} \]  

(9)

and its cumulative density function (cdf) is the sum of the probability density function in all the points below a given value:

\[ F(\mu, x) = \sum_{i=0}^{x} \frac{e^{-\mu} \mu^i}{i!} \]  

(10)

The parameter \( \mu \) represents the mean of the distribution. In our case, it will represent the number of expected changes.

We will use the properties of the Poisson distribution as follows: we are expecting to have \( \mu \) occurrences in a given period of time. Therefore, initially we will accept a change with a very high probability, but as the number of accepted changes increases and is close or over the expected number, they will be more likely to be rejected. This is easily modeled using the cumulative density function \( F(\mu, x) \) as a discard probability: this discard probability will be very low at first, and as we get closer to the mean or past it, it will get bigger and bigger, until a moment where it will be close to 1 (that means, all the occurrences will be rejected). In figure 2 we can see how the discard probability increases as the number of accepted changes gets bigger.

![Discard probability on the Poisson-based discard algorithm](image)

**Fig. 2.** Discard probability on the Poisson-based discard algorithm.

### 4. EXPERIMENTAL FRAMEWORK

We have examined and evaluated the proposed false alarm rejection strategies on the evaluation data set of the Spanish Parliament Sessions defined for the 2006 TC-STAR Automatic Speech Recognition (ASR) evaluation campaign [5]. This database contains Cortes Spanish Parliament speeches. Table 1 summarizes the data available in the dataset used.

### 5. EXPERIMENTAL RESULTS

#### 5.1. Evaluation measures

To evaluate the performance of the speaker segmentation system we use the measures of Precision \( P \) (% of detected points which are genuine change points), Recall \( R \) (% of detected speaker change points) and F-score \( F \). \( P \) measures the number of insertions in function of the number of changes found, \( R \) is the same but attending to the number of deletions, and \( F \) is a combination of both parameters. The bigger these quality parameters are, the better the system is. The expressions used to measure these parameters are

\[ P = \frac{c}{c + i} \times 100 \]  

(11)

\[ R = \frac{c}{c + d} \times 100 \]  

(12)

\[ F = \frac{(1 + \beta^2)PR}{\beta^2P + R} \]  

(13)

where \( c \) is the number of target changes, \( i \) is the number of insertions (changes that are not real ones) and \( d \) is the number of deletions (target changes that were not found). We select \( \beta = 1 \), what means that the F-score gives the same importance to \( P \) and \( R \).

Given the aim pursued in this work, we are also going to pay attention to the number of total changes found by the two systems, being better the one with the lower number of them.

#### 5.2. Results

Table 2 shows detailed results of the three proposed false alarm reduction strategies and the results of the baseline system, which does not use a false alarm rejection technique. The values observed are the number of deletions (d), the number of insertions (i), the precision (P), the recall (R), the F-score (F), the number of obtained segments (s) and the number of wrongly discarded changes (w) by the rejection strategy.

These results were obtained after adjusting some parameters: on the adaptive technique, the maximum threshold was set to 300; on the uniform-based system the threshold was set to 500 and the discard probability to 0.8; and on the Poisson-based the threshold was set to 500 and the mean (expected number of segments) to 15.

### Table 1. Summary of the datasets used in the experiments.

<table>
<thead>
<tr>
<th>Total length</th>
<th>Segment length</th>
<th>Speaker changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eval06</td>
<td>115 min</td>
<td>2 sec - 10 min</td>
</tr>
</tbody>
</table>

The audio signal is characterized by 12 mel-frequency cepstral coefficients (MFCC) extracted every 10 ms using 25 ms Hamming windows. Then these cepstral features are augmented by the log-energy. The DCTS and BIC change detection stages use only the 12 MFCCs and the log-energy as features. In the speech/non-speech classification and the gender classification modules the first and second derivatives of this feature vector were also considered.

The speech, non-speech, male and female models were 64 diagonal Gaussian Mixture Models (GMM) directly trained on data extracted from the train corpus, defined for the TC-STAR evaluation, using the Expectation-Maximization (EM) algorithm.
Fig. 3 shows the results obtained by the uniform-based system as a function of the discard probability (threshold). It can be seen that the value of 0.8 makes a tradeoff between precision and recall. When a higher value is chosen, precision grows but recall is reduced, and with lower values the recall grows but also the precision decreases. In Fig. 4, we show the influence of the value given to the expected number of changes on the Poisson-based rejection approach. With very low values, we have a better precision but a worse recall, but as the number of expected changes grows, precision decreases but recall obtains its maximum value and becomes constant. This indicates that the number of expected changes has to be carefully selected, as a big one will improve recall because less changes will be discarded, but it will have a really bad influence in precision, greatly increasing the number of insertions.

Comparing the results in Table 2 it can appreciate that the three false alarm reduction techniques achieve their goal, as the number of false alarms with respect to the baseline is highly reduced, and therefore the precision increases.

Table 2. Comparison between the false alarm rejection strategies.

<table>
<thead>
<tr>
<th>System</th>
<th>c</th>
<th>d</th>
<th>l</th>
<th>P</th>
<th>R</th>
<th>F</th>
<th>s</th>
<th>w</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>42</td>
<td>6</td>
<td>48</td>
<td>57.46</td>
<td>85.59</td>
<td>65.99</td>
<td>242</td>
<td>-</td>
</tr>
<tr>
<td>Adaptive</td>
<td>42</td>
<td>6</td>
<td>25</td>
<td>66.98</td>
<td>85.59</td>
<td>73.17</td>
<td>197</td>
<td>5</td>
</tr>
<tr>
<td>Uniform</td>
<td>42</td>
<td>8</td>
<td>9</td>
<td>80.97</td>
<td>81.75</td>
<td>81.22</td>
<td>145</td>
<td>7</td>
</tr>
<tr>
<td>Poisson</td>
<td>42</td>
<td>7</td>
<td>12</td>
<td>76.21</td>
<td>83.09</td>
<td>79.05</td>
<td>146</td>
<td>5</td>
</tr>
</tbody>
</table>

Although we want to reduce the number of false alarms, we should try to keep the number of incorrectly rejected change-points as low as possible. The adaptive technique keeps the number of incorrectly deleted change-points obtained with the baseline, but the other two techniques have more deletions of true change-points, the worst being the uniform-based one. Nevertheless, the uniform-based system obtains the lowest number of false alarms, raising its precision and obtaining the best F-score of the three systems. Attending to the number of segments, the performance of the uniform-based technique and the Poisson-based were both good, as they reduced in a big amount the number of obtained segments. That suits the other results, as when we make less segments, we will have less insertions. Finally, it can also be seen that the uniform-based method makes a higher number of mistakes, which would explain why the number of deletions increases with this approach.

6. CONCLUSIONS AND FUTURE WORK

Two novel false alarm reduction techniques are proposed and compared. The experimental results confirm the validity of these approaches as they achieve the expected result of reducing the number of false alarms. Those results suggest that the uniform-based approach is better when we want to have the chance to select the importance we want to give to false alarms, adjusting the value of the discard probability. On the other hand, Poisson-based reduction technique is useful when we have an idea of the number of speaker turns that we would expect, so we can select the proper value for the mean of the distribution, which is equivalent to the number of expected speaker segments.

In future work, it would be interesting to make a fusion of the different false alarm reduction strategies, in order to take advantage of the positive side of each one. In addition, we will try to combine all the available information sources (such as the $\Delta BIC$ value, the CLR and the proposed discard probability) in order to improve the results. We will keep working on the idea of modeling the sequence of change-points found in an audio stream as a Poisson process because of its potential shown in other areas, as it might be successfully extended to speaker change detection.

7. REFERENCES


