Speech Cohesion for Topic Segmentation of Spoken Contents

Abdessalam Bouchekif1 2, Géraldine Damnati1, Delphine Charlet 1

1 Orange Labs, 2 avenue Pierre Marzin 22307 Lannion, France
2 LIUM, Université du Maine, Le Mans, France

{abdessalam.bouchekif,geraldine.damnati,delphine.charlet}@orange.com

Abstract

In this paper, we introduce the notion of speech cohesion for topic segmentation of a spoken content. The aim is to integrate speaker information and lexical information within a single cohesion value. Based on a lexical cohesion system, we propose an approach that directly integrates the speaker distribution when processing the cohesion. A potential boundary is effective if the joint distribution of terms and speakers is different enough from one side of the boundary to the other. Beyond speaker distribution, we also propose to take into account speaker identification and to confront speaker identities to entities mentioned in the spoken content in order to reinforce cohesion of a topic segment. Experiments run on three corpora of various Broadcasts News forms collected from 9 French TV channels, show a significant improvement in the overall topic segmentation process.

Index Terms: Topic segmentation, speech cohesion, Speakers distribution, speaker identification.

1. Introduction

Topic segmentation (TS) is a task which consists in finding thematically homogeneous fragments: i.e. talking about a single subject. Knowing the transitions from one topic to another can be used in many fields such as automatic summarization, indexing and information retrieval. Three categories of cues or features have been explored in the task of TS for TV Broadcast News (TVBN): lexical, acoustic and visual cues. Lexical cues are based on the exploitation of the automatic transcription. Main lexical approaches rely on the notions of lexical cohesion introduced in [1] and lexical chaining [2]. Acoustic cues can be pause duration: news topics are separated by significant pauses or music clips. Anchor information is shown to be strongly correlated with story boundaries [3]. Xie et al. [4] exploit Speech type based on the assumption that news story usually starts from a clean speech (in the studio), following a noisy environment (street, plants, ...). Visual cues (such as a news title, caption, logo detection, shot detection, etc...), can also reveal topic shifts but they heavily rely on editorial rules [5]. In this paper, we focus on the audio, not exploiting any information from the video. Our approach can apply to any spoken broadcast content, from TV or radio.

Various methods for TS based on lexical cohesion have been proposed relying on similarity computation. TextTiling algorithm [1] measures lexical cohesion between adjacent pairs of blocks using a sliding window along the show. A high value implies that there are terms in common between these two blocks and that they are likely to belong to the same subject. A low value indicates a topic change. C99 [6] and MinCut [7] algorithms compute pairwise similarity between every pair of sentences. These values are exploited in an indirect way across a rank matrix and an undirected graph respectively. Our approach is inspired from TextTiling algorithm with an original term weighting scheme for cohesion computation and an improved boundary detection algorithm [8]. In this paper, we introduce speech cohesion, a new notion that generalizes lexical cohesion by jointly taking into account lexical and speaker information. In a first step, we integrate speaker distribution with the cohesion computation and in a second step we propose to exploit speaker identification along with speaker mention. Speaker role has been exploited in several studies. In previous work, we showed that detection of the anchor speaker can yield significant improvements whether when integrated in the boundary selection algorithm [3] or in the term weighting scheme [8]. Indeed the anchor speaker is usually in charge of introducing new topics. However this information is not sufficient in itself as a new topic can be introduced by another speaker, and on the opposite, the anchor speaker can occur several times during a single topic. Hence we propose to exploit speaker distribution in more details. Most of TS that exploit speaker informations have adopted supervised framework (i.e using classifiers). For example, [4] exploit speaker segmentation via a binary features (speaker change / no change) to designate a speaker change. Dumont et al. [9] use the results of speaker diarization by adding the index of the speaker in each observation. In this case, the possible values for this descriptor is the number of users in the broadcast.

In this paper, we propose a method which directly integrates the distribution of speakers in the cohesion computation with an unsupervised approach. Indeed, TVBN shows usually contain an anchor, reporters and guests. If the anchor is generally present along the show, interviewed guests are likely to speak only in a single subject. Furthermore, if we are able to identify speakers, it is likely that their name is uttered before or after their intervention. Combining speaker identification and spoken name detection can further reinforce the cohesion of topically coherent segments. The concept of cohesion applied to terms distribution can be extended to speakers distribution and generalized to the new notion of speech cohesion.

The rest of this paper is divided as follows. The topic segmentation algorithm that is used in our work is presented in section 2. We describe in section 3, the integration of speaker information in the cohesion computation. Finally in the last section, we introduce a new metric for evaluating segment retrieval and we report the detailed of the properties of our databases and experimental results.

2. Topic segmentation algorithm

Our topic segmentation baseline system is based on the analysis of lexical distribution. A similarity measure is computed between each pair of adjacent blocks. In automatic transcrip-
tion, sentence boundary detection is not a trivial task. There are neither punctuation nor capital letters. Rather than sentences, units are breath groups (BG) which are sequences of words between two pauses in a speech turn. Pauses are automatically detected by the Automatic Speech Recognition engine. Similarity is computed using a sliding window of size $2 \times K$ between adjacent blocks of $K$ BGs along the show. The similarity values constitute a cohesion curve, each point of the curve being the value associated to a potential boundary. A boundary selection algorithm is applied in order to detect disruptions on the curve.

### 2.1. Intra-document weighting

Term weighting is intended to reflect how important a word is to a document. We perform intra-document term weighting without any external information sources. In [7] TF-Idf weights are computed from a uniform partition of a show into $N$ chunks, each chunk representing a document in the classical TF-Idf framework. In [8] we have proposed two approaches in order to outperform the state-of-the-art uniform chunk partition. In the first one, weights are estimated iteratively. Topic segmentation obtained at a given iteration provides a set of documents from which weights are re-estimated for the next iteration. The second approach makes use of structural information provided by anchor speaker turns detection. In this paper, we use the second approach. The beginning of each anchor speaker turn is considered as the beginning of a new chunk. It is interesting to notice that anchor information itself is not sufficient because in our corpora, the first BG of an anchor speaker does not necessarily correspond to a boundary and what’s more a new topic is not always introduced by the anchor speaker. But this approach provides a good trade off for partitioning the show for term weighting. Speaker role analysis is performed following the multi-stage process described in [10]. A term $t$ in a breath group $x$ will be associated to a weight $w(c(x), t)$ depending on the chunk $c(x)$ in which it occurs.

$$w_{TF-IDF}(c(x), t) = TF(c(x), t) \times IDF(t)$$

where

$$TF(c(x), t) = \frac{n_t}{\sum_{j=1}^{N} n_j}$$

and

$$IDF(t) = \log\left(\frac{N}{n_t}\right)$$

$n_t$ is the number of chunks containing term $t$.

$N$ is the total number of chunks.

### 2.2. Similarity Computation

For a given potential boundary between two adjacent blocks $b_j$ and $b_{j+1}$ of $K$ BGs, we use the cosine similarity which measures the proximity between representation vectors $V_j$ and $V_{j+1}$ of $b_j$ and $b_{j+1}$ respectively. The vector coordinate of term $t$ in block $b$ is a weighted value $v(b, t)$. In our representation, there is not one unique weight for a term $t$ in block $b$ as breath groups of the block may not all belong to the same chunk. Hence, the block level weighted value $v(b, t)$ is obtained by summing up weighted values for each BG $x$ contained in the block.

$$v(b, t) = \sum_{x \in b} (f_{x,t} \times w(c(x), t))$$

where $f_{x,t}$ is the frequency of term $t$ in BG $x$.

For a given potential boundary between blocks $b_j$ and $b_{j+1}$, the similarity is $\text{cohesion}(j) = \text{cosine}(V_j, V_{j+1})$.

$$\text{cohesion}(j) = \frac{\sum_t (v(b_j, t) \times v(b_{j+1}, t))}{\sqrt{\sum_t (v(b_j, t)^2) \times \sum_t (v(b_{j+1}, t))^2}}$$

### 2.3. Boundary selection

The cohesion value associated to each potential boundary can be plotted as a cohesion curve from which boundaries can be detected. The classical approach [1] consists in detecting valleys in the curve, as the lowest point between two peaks (local maxima), and selecting $x_j$ as a boundary if the valley depth $d(j)$ is above a given threshold. The depth is computed as the sum of the differences between the lowest point value and the left and right peaks values. Directly working on valley depth is not always optimal [3] as some topic segments may not contain enough term repetitions to yield high lexical cohesion values. Similarly, directly searching for low lexical similarity values is not optimal (some topic shifts between closely related topics can result into average lexical similarity values). We use a combination of those measures through linear interpolation. For a potential boundary $j$ the score is given by:

$$\text{score}(j) = \lambda(1 - \text{cohesion}(j)) + (1 - \lambda)\text{depth}(j)$$

We fixed $\lambda$ to 0.75. This dissimilarity score emphasizes low values of cohesion which are also local minima.

Rather than simply applying a threshold to determine high values of this score, we proposed an iterative splitting algorithm (for more details see [8]).

### 3. From lexical cohesion to speech cohesion

#### 3.1. Speaker diarization

Speaker diarization is the task that segments speech into speaker turns, and that clusters together all the speech turns of a same speaker. Speakers of the document are then assigned a unique label, e.g. speakerA, speakerB,... throughout the document. The possible errors in speaker diarization can come from a segmentation error (2 speaker turns merged in one), or from clustering errors (2 speech turns from the same speaker attributed to different clusters, or 2 speech turns from different speakers in the same cluster). The system used here is presented in [11]: it is a sequential system using firstly Bayesian Information Criterion and then Cross-likelihood Criterion.

#### 3.2. Speaker Identification

Speakers are identified thanks to a multimodal fusion process which exploits acoustic speaker recognition and Optical Character Recognition. In most cases, the name of the speaker is overlaid in the image. The fusion process exploits speaker clustering to propagate the speaker identity to all his speech turns. The whole process is explained in details in [12], except that the GMM used for speaker recognition is replaced here by I-Vector [13]. The speaker identification system is part of the multimodal person recognition REPERE challenge [14].

#### 3.3. Speech cohesion

Our purpose is to define a new cohesion value that reflects several information dimensions conveyed by speech. In fact, a topic segment is characterized by a particular distribution of words (lexical cohesion) but also by a particular distribution of speakers (speaker cohesion). If the anchor speaker prevails all along the show, some reporters or some guest speakers only occur during a given topic. We propose an approach that directly integrates speaker distribution within a unified cohesion computation. As a preliminary step, we proceed a synchronization step between BGs determined by the transcription system [15] and speaker turns obtained by the speaker diarization system.
Then, we define a new representation vector for a given block \( b \). If the show contains \( N_T \) different tokens, the representation vector of a block \( b \) for lexical cohesion computation is composed of \( N_T \) components \( v(b,t) \) defined as the weighted frequency of each token \( t \) in block \( b \), as defined in equation 2. Similarly, if the show contains \( N_S \) different speakers, the representation vector of a block \( b \) for speaker cohesion computation is composed of \( N_S \) components \( v(b,s) \) defined as the weighted frequency of each speaker \( s \) in block \( b \). The notion of frequency for a speaker is defined as a function of the number of tokens uttered by a speaker in a BG \( x \).

\[
v(b,s) = \sum_{x \in b} (f_{x,s} \times w(c(x), s))
\]

where the frequency \( f_{x,s} \) of speaker \( s \) in BG \( x \) is defined as follows:

\[
f_{x,s} = \frac{1 + n_{b,x}}{1 + \max} \]

where \( n_{b,x} \) is the number of tokens uttered in BG \( x \), and \( \max \) is the maximum number of tokens among all breath groups of the show. The weight \( w(c(x), s) \) associated to speaker \( s \) for BG \( x \) is computed following the same method as for tokens. In this case weighting allows to reduce the contribution of regular speakers (typically the anchor speaker) while enhancing the contribution of some less frequent speakers. Finally the representation vector of a block \( b \) for the computation of speech cohesion is simply the concatenation of the \( N_T \) lexical components and the \( N_S \) speaker components. Hence, both information sources are jointly taken into account for the computation of the cohesion value \( \text{cohesion}(j) \). We have chosen this definition for \( f_{x,s} \) as an approximation of speech duration focusing on useful tokens (i.e. after removing function words and words above a given confidence threshold, as is usually done for lexical cohesion computation). Grounding the frequency by 1 prevents zero values for breath groups containing no "useful" tokens. Additionally, normalizing by \( 1 + \max \) guarantees a more balanced contribution of lexical and speaker information. Several normalization approaches have been experimented and this one provided the best performances.

When speaker identification is available, labels of speakers \( s \) are replaced by their identity. Speech cohesion can be additionally reinforced by including spoken name detection information. In fact, a reporter is usually introduced by the anchor speaker, and the name of a guest is usually mentioned by the anchor or the reporter, preceding or following the guest’s utterance. Spoken names detection and normalization is performed following the REPERE challenge framework using the approach described in [16]. Provided that mentioned names and speakers are identified following the same standardization rules (here Firstname_LASTNAME), both information can be combined within the speech cohesion computation. Hence a person name can contribute to speech cohesion through the number of words uttered by this person (within speaker cohesion) or by the number of times this person name is mentioned (within lexical cohesion).

### 4. Experiments and Results

#### 4.1. Corpora

Experiments are carried out on three sets of shows. The first one named Multi Channels (MC) is composed of 33 French TVBN shows broadcasted between October 2008 and January 2009 from 7 different channels\(^1\), for an average duration of 22 minutes. The second set is composed of 23 shows from a new channel, D8 (during October 2013), for an average duration of 13 minutes. Similarly to [17] the first and the last topics of a show are discarded when they correspond respectively to the titles presentation or summary. The third set of shows is composed of BFMStory shows, as part of the REPERE challenge corpus. BFMStory is a talk-show that includes Broadcast News (BN) portions, ads, interviews and debates. Here we focus on BN portions that have been manually isolated. 33 talk-shows (recorded between May 2011 and September 2012) yield 72 BN portions, for an average duration of 7 minutes.

Automatic transcription is performed by [15]. It achieves 16.1% word on the MC corpus. Manual transcriptions are not available for D8 corpus and only partially for the BFMStory corpus, preventing ASR performance evaluation. On the BFMStory corpus, the additional components of speaker diarization and speaker identification have been evaluated in the REPERE challenge [14], and give state-of-the-art performances, respectively 10.3% for the diarization error rate, and 12.5% for the speaker identification error rate, decomposed into 9% confusion, 2% insertion and 1.5% missed detection.

From an applicative point of view, we assume that long segments are more important to retrieve than short ones, as they convey more information on their topic and are more likely to be re-used after their first live broadcast. Thus, we make distinct evaluations, according to the duration of the reference segments. When analyzing the distribution of the duration of the segments, it appears that they can be easily divided in two sets, where the threshold between short and long segments is set to 30s. Table 1 describes the repartition of short and long segments in the different corpora. For evident reasons, short segments are more difficult to retrieve than long segments and the BFMStory corpus is more difficult to process than the D8 corpus.

#### 4.2. Evaluation Metrics

Usually seen as a topic boundary detection task, topic segmentation can also be seen as a homogeneous segment detection task, where the purpose is to provide automatic segments that best fit reference segments. Thus we use two evaluation metrics, one based on boundary detection and the other one based on segment retrieval.

##### 4.2.1. Boundary detection evaluation

Performances are measured in terms of recall and precision by comparing time information associated to hypotheses and reference boundaries. As frequently found in the literature, an interval of 10s before and after a boundary hypothesis is tolerated in order to decide if it is correct.

##### 4.2.2. Segment retrieval evaluation

We propose an evaluation measure that reflects the quality of topic segmentation, in terms of reference segments retrieval.

For each reference segment, the evaluation process searches for the test segment which best fits the reference segment, i.e.

---

\(^1\)TF1, France2, France3, LCI, France24, Arte, M6
which has the maximal temporal coverage of the reference segment. Thus, each reference segment $\text{ref}_i$ has a corresponding test segment $\text{test}_i^{\text{match}(\text{ref}_i)}$, which can be seen as the system’s response to the task of retrieving $\text{ref}_i$.

To measure the quality of the system’s response for this specific reference segment $\text{ref}_i$, we define $\text{cover}_{\text{ref}_i}$ as the percentage of duration of the reference segment which is correctly retrieved by the test segment. It can be seen a temporal recall rate for the reference segment $\text{ref}_i$. Conversely, we define $\text{covertest}_i$ as the percentage of duration of the test segment which corresponds to the reference segment, which can be seen as a temporal precision rate for test. Pursuing this analogy, we define $\text{harmoniccover}_i$ as the harmonic mean of $\text{cover}_{\text{ref}_i}$ and $\text{covertest}_i$. For a given corpus, the overall $\text{cover}$, $\text{covertest}$ and $\text{harmoniccover}$ are defined as the average over all reference segments of the individual corresponding values.

The main advantage of these metrics is to measure the retrieval quality for each reference segment separately (independently of any tolerance margin definition). It thus enables deeper performance analysis, by computing the average metrics in function of some properties of the reference segments (e.g. length, signal to noise ratio...). In this paper we will compare retrieval performance for long and short segments.

### 4.3. Experimental results

#### 4.3.1. With speaker identification

Table 2 illustrates the performances of our system with and without taking into account speaker distribution. Speaker distribution alone doesn’t provide sufficient performance but its combination with lexical distribution within speech cohesion gives interesting results. Indeed, taking account both of these informations increase the boundary detection F-measure by +2.4% and +7.1% absolute for MC and D8 respectively. Similar overall performances are observed with evaluation relative to segment retrieval. After analysis of the results achieved, we have seen that adding label of speaker in similarity computing allows to refine the segmentation even if there is little repetition of terms within topic segment. The results show that the combination yields the best results on D8 corpus. From the analysis of these corpora, we found that this difference is mainly due to the proportion of long segments, which is larger in the D8 corpus. Short segments are more numerous in MC corpus, and prevail in the BFMStory corpus. This type of segment is not favorable to our system based on redundancy. Indeed, topics under 30s generally correspond to short news read only by the anchor or other speaker, which are difficult to isolate. For this reason, few topic changes are detected. On the other hand long segment retrieval is particularly favored by speech cohesion.

#### 4.3.2. With speaker identification

Finally, we have run a set of experiments for BFMStory corpus. The particularity of these shows is that they contain much more short segments than MC and D8 corpora (see table 1). In addition to speaker diarization (section 3.1) which gives anonymous segment labels (i.e speakerA, speakerB, ...), we have also used speaker identification (section 3.2) which consists in labeling those segments with their speaker name, and spoken name detection (section 3.3). Using this information reinforces the cohesion between what is said and who says it. Even though the impact of speaker distribution and identification is poor for the overall BFMStory corpus, due to the important amount of short segments, we can see that performances are improved for long segments retrieval, starting from a harmonic cover of 77.8% with lexical cohesion only, to 79.2% with speaker identification. For BFMStory long segments $\text{cover}$ is equal to 85.2% and $\text{covertest}$ is equal to 82.3% meaning that on average, long segments are retrieved with a 85.2% coverage of their duration.

### 5. Conclusion

This paper proposes an approach to integrate the distribution of speakers and lexical cohesion information within a new paradigm for topic segmentation: speech cohesion. The simultaneous use of both information sources improves significantly the results in terms of boundary detection precision and recall. We have proposed a new metric that reflects the quality of the system as a segment retrieval system, in terms of temporal segment coverage. This metric reveals that speech cohesion improves the performance of the system in particular for long segments retrieval. However, performance of the system is still to be improved in the case of shorter segments, these should be subject to special processing.

<table>
<thead>
<tr>
<th>Condition</th>
<th>Boundary detection</th>
<th>Segment retrieval</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Recall</td>
<td>Precision</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------</td>
<td>-----------</td>
</tr>
<tr>
<td>Multi Channels</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Speaker cohesion</td>
<td>34.6</td>
<td>35.8</td>
</tr>
<tr>
<td>Lexical cohesion</td>
<td>63.6</td>
<td>63.2</td>
</tr>
<tr>
<td>Speech cohesion</td>
<td>66.5</td>
<td>65.1</td>
</tr>
<tr>
<td>D8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Speaker cohesion</td>
<td>40.5</td>
<td>48.6</td>
</tr>
<tr>
<td>Lexical cohesion</td>
<td>77.4</td>
<td>58.0</td>
</tr>
<tr>
<td>Speech cohesion</td>
<td>77.1</td>
<td>69.9</td>
</tr>
<tr>
<td>BFMStory</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Speaker cohesion</td>
<td>46.9</td>
<td>10.1</td>
</tr>
<tr>
<td>Lexical cohesion</td>
<td>59.2</td>
<td>61.3</td>
</tr>
<tr>
<td>Speech cohesion</td>
<td>56.9</td>
<td>62.4</td>
</tr>
<tr>
<td>Speech cohesion + id</td>
<td>57.0</td>
<td>63.7</td>
</tr>
</tbody>
</table>

Table 2: Topic segmentation evaluation

![Figure 1: Example of segment retrieval evaluation](image-url)
6. References


