Topic Segmentation of TV-streams by mathematical morphology and vectorization

Vincent Claveau[1], *Sébastien Lefèvre*[2]

[1]IRISA-CNRS, Rennes, France
[2]Valoria, University of South Brittany, France
vincent.claveau@irisa.fr, sebastien.lefevre@univ-ubs.fr

Abstract

A fine-grained segmentation of Radio or TV broadcasts is an essential step for most multimedia processings. Applying segmentation algorithms to the speech transcripts seems straightforward. Yet, most of these algorithms are not suited when dealing with short segments or noisy data. In this paper, we propose a new segmentation technique inspired from the image segmentation field and relying on a new way to compute similarities between candidate segments. This new topic segmentation technique is evaluated on two corpora of French TV broadcasts on which it largely outperforms other existing approaches from the state-of-the-art.

Index Terms: topic segmentation, TV stream segmentation, vectorization, mathematical morphology, watershed transform.

1. Introduction

Topic segmentation is of high interest in Multimedia IR. Indeed, it is needed to perform automatic structuring of TV streams, a keystone for every processing of such streams, which is still done manually in national archive agencies like the French INA. A way to obtain this structuration is to first transcribe the audio tracks of the TV streams into textual data, and then perform the topic segmentation from textual data to split the streams into semantic units (e.g., reports).

In this paper we address the problem of topic segmentation of speech in this applicative framework based on a twofold contribution. First, our topic segmentation system is based on Mathematical Morphology principles that are usually used for image segmentation. Secondly, a key component for this approach is the calculation of the similarity between 2 successive possible segments; in this paper we propose to use a new technique, called *vectorization* that we recently introduced in the information retrieval field.

The paper is organized as follows. We first present state-of-the-art approaches used for topic-segmentation. We then show that topic segmentation and image segmentation have common characteristics (Sec. 3). From this observation we build a topic segmentation method based on the watershed transform, a common morphological tool that identifies segments or regions within a topographic surface. We suggest to build this topographic surface with the help of vectorization which we think is especially suited when dealing with small segments or noisy data as TV streams (Sec. 4). Experiments performed on two TV broadcast corpora are presented and discussed (Sec. 5). Finally, Sec. 6 concludes this work and provides future research directions.

2. Related work in topic segmentation

Several approaches have been applied to topic segmentation. Some of these ones rely on some particularities of the document format, or on the detection discourse markers [1]. Conversely, the other family of approaches examine the content of the document to detect the change of topic. This is also the approach adopted in our system. In the following, we briefly present some of the best-known among the state-of-the-art.

The segmenting process of SEGMENTER [2] relies on a representation of the text as weighted lexical chains. Finding the boundaries is thus equivalent to partitioning the resulting graph. The two approaches in DOTPLOTTING [3] and C99 [4] differ in the way the content is represented, but both rely on the computation of similarities between the candidate segments and then on a clustering based on the resulting similarity matrix. The computation of similarity is also at the heart of the TEXT-TILING system [5], in which a sliding window is used to compare the content before and after each possible boundary. The similarity measure used is inspired from the information retrieval domain (for instance, TF or TF-IDF), and the final boundaries are searched among the places in which the lexical cohesion reaches a significant local minimum. More recently, Utiyama et Isahara [6] proposed to use a statistical approach based on an hidden Markov model. Here again, the lexical cohesion is measured classically with the help of language modeling.

All these different approaches have been compared [4, 7, respectively on English and French]. It is worth noting that all these approaches rely on the word repetition to compute some kind of similarity in order to decide if the topic is changing or not. Therefore, it has been noticed that dealing with segments with very few common words, like short segments, is very challenging for this family of approaches. In order to limit the impact of this problem, several authors have proposed to use existing lexical resources or to build them. For instance, Guinaudeau *et al* [8] integrated semantically related terms to the segmentation model of [6] in order to extend the description of the possible segments. Our approach, thanks to the properties of the vectorization, is expected to be more suited for this kind of problem (cf. Sec. 4).

3. Topic segmentation as morphological segmentation

3.1. Morphological segmentation

Mathematical morphology is both a rich theoretical framework and a complete toolbox mostly used in the image processing community. Particularly, it has been extensively used for image segmentation, which aims at splitting an input image into

a set of uniform regions given a predefined uniformity criterion (intensity or colour, texture, etc.). The most famous morphological method for image segmentation is certainly the watershed transform.

We recall very briefly the principle of watershed-based segmentation [9, for a comprehensive presentation]. The image I to be segmented is first represented as a topographic surface. Watershed lines identified on this surface are then associated to region frontiers resulting from the segmentation process. One common way to implement it is to simulate the progressive flooding of the surface starting from its local minima, and then to build dams to avoid merging water from two different catchment basins. At the end of the process, dams correspond to the watershed lines or, in other words, to the region frontiers (see Figure 1).

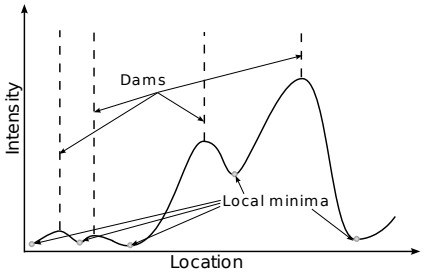

Figure 1: Example of a watershed in 1-D: the altitude of each pixel p is defined by its intensity $I(p)$ in the image to segment.

Most often, this approach is not directly applied on the image I to be segmented. Before applying the segmentation, an image transform is rather performed as a preprocessing in order to highlight values of edge pixels and to lower pixel values in homogeneous areas. A gradient (noted ∇I hereafter) is thus usually computed to enhance transition areas (which generally correspond to object frontiers). Various gradient computation methods exist. Its choice is of high importance, since it will directly influence the segmentation result produced by the watershed method.

3.2. From image to text

The analogy between image and text segmentation can be drawn very simply. The pixel is the base element in the image and is described by its greylevel or color/multispectral values. Its equivalent in texts is the sentence (or sometimes the paragraph) which is described by the words it contains. In our framework of video segmentation, our texts are obtained from automatic transcription. Thus, the transcribed utterances are the minimal units of the text (i.e., they are equivalent to image pixels) and topic breaks will be sought between them.

Besides, our texts are flows of utterances. They are then represented as 1-D signals, while images are most often 2- or 3-dimensional. However, nothing prevents the watershed technique to be applied on a single dimension as shown in Fig. 1. Thus our approach relies on a gradient computed on the sequence of utterances, and topic breaks are identified using the watershed transform. Gradient computation, which is a key step of the segmentation process, is detailled in Sec. 4. The watershed technique used here is the standard one described previously. We have only included a gradient smoothing step to remove irrelevant local minima.

4. Gradient computation using vectorization

4.1. Vectorization principles

Vectorization is an embedding technique which aims to project any similarity computation between two documents (or one document and one request in the context of IR) in a vectorial space. It has been introduced and experimented in a standard IR scenario [10] where it has shown to provide both a low complexity and accurate results. We recall here its main characteristics.

Its principle is relatively simple. For each document of the considered collection, it consists in computing with an initial similarity measure (e.g., standard similarity measure used in IR), whatever it is, some proximity scores with m pivot-documents. These m scores are then gathered into a m-dimensional vector representing the document (cf. Fig. 2).

Comparing two documents (or a document and a request) can then be performed in a very standard way in this vectorial space (e.g., using a L_2 distance). Many algorithms are available to compute or approximate very efficiently such distances.

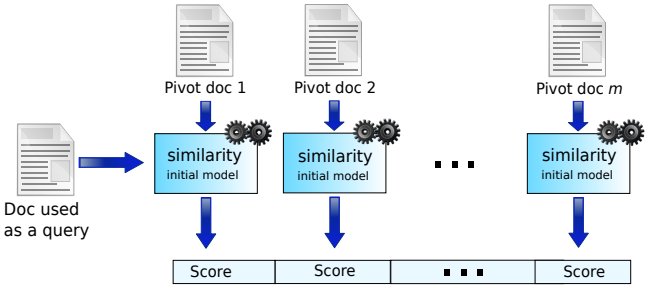

Figure 2: Vector design from pivot-documents

More formally, we note Vect$(D, \mathcal{P}, Sim)$ the vector representing the document D built from the initial similarity measure Sim on pivot-documents $\mathcal{P}$. For instance, Vect$(D, [P_1, P_2, P_3]$, TF.IDF/cosine$)$ is a 3-dimensional vector; its first component is the similarity score between the document D and the pivot-document P_1 returned by a system using TF.IDF representation associated to the cosine distance measure (which corresponds to a very standard way to compute similarities in the IR field; TF and IDF respectively stand for Term Frequency and Inverse Document Frequency [11, for details]), and so on for the next components.

4.2. Properties

It is important to notice that vectorization results in a change of the representation space, contrary to existing works consisting rather of a dimension reduction or a distance approximation (e.g., [12]). This space transform offers several nice properties which will be discussed here.

The first interest of this embedding is to reduce complexity when the initial similarity computation may be computationally expensive (e.g., some graph comparison computations used in complex IR systems). In a IR context, vectors associated with each document may be built offline, and when a request has to be processed, we only need to compute its similarity with the m pivot-documents rather than to do it with all documents in the collection. This property is nevertheless not useful in the context of a segmentation task.

The second nice property comes from the fact that two documents will be considered as similar if they are similar to the same pivot-documents. This indirect comparison, or second-order affinity, let us compare two textual documents which do not share any common word. This property will be helpful in our segmentation task. Indeed, it will solve the problem brought by the lack of repetition between utterances. This problem is particularly noticeable when the segments to be compared are of short duration (i.e., they will contain less words, and thus will share only a few words in common in the best case, and no common word in the worst case).

4.3. Usage

A gradient is computed between each utterance. In other words, we compute the similarity using the vectorization principle between previous and next utterances. Let us note that we do not compare only the previous to the next utterance, but we also consider the n previous ones vs. the n next ones (similarly to TEXT-TILING, a common approach for topic segmentation).

In experiments described in the following section, the initial similarity measure used in the vectorization process is a L_2 distance associated with a weighting of utterances by $\sqrt{TF}$. It means that we first represent each breath group by a sparse vector in which each dimension represent a word; the value for this dimension is the square root of the number of occurrences of the word in the breath group. The same is done for the pivot document. The distance between the breath group vector and the pivot vector is computed with a L_2 distance; the resulting value forms one of the dimensions of the new vector.

Similarly to some image gradient computation methods, we give more importance to close utterances and less importance to utterances which are far to the candidate edge. This is ensured through a simple convolution with a kernel (e.g., Gaussian kernel). Let us notice that the way the convolution is applied depends on the way the documents are represented in the initial model of similarity computation. With the vectorial representation used in our experiments, this convolution is simply taken into account: when computing $\sqrt{TF}$, the occurrence of a word counts for one in the breath group which is the closest from the candidate edge, but counts for less when considering an occurrence from a breath group further of the candidate edge. In practice, a linear penalty is applied. From now we will write $\mathcal{C}_{prev}(i)$ (respectively $\mathcal{C}_{next}(i)$) the result of the convolution operator applied on utterance i and those which are preceding (respectively following) it.

Formally, the gradient is thus defined by:

$$\nabla(i) = L_2(\text{Vect}(\mathcal{C}_{prev}(i-1), \mathcal{P}, \sqrt{TF}/L_2), \text{Vect}(\mathcal{C}_{next}(i), \mathcal{P}, \sqrt{TF}/L_2))$$

Pivot-documents we are using are simply sequences of utterances built from random splits of the considered broadcast.

In the experiments reported below, utterances are represented by their starting time. For a given time index, the higher the gradient is, the more important the dissimilarity between previous and next groups is. In other words, significant local maxima of gradient values indicate a topic break.

5. Experiments

5.1. Experimental data

Our experiments are performed on two French TV broadcast corpora for which the topic segmentation is of high interest. The first corpus is a set of 60 TV news of the France 2 channel (called *News* further). Each of these sample has been broadcasted in the beginning of 2007 and is 40 minutes long. The second corpus is made from TV reports: 12 samples of *Envoyé spécial* (2008, 2 hours long each), and 16 *Sept à huit* (2008, 1 hour long each). This corpus is called *Reports* in the following experiments.

These corpora [8] have different properties in terms of number and duration of topic segments. Thus, it allows us to evaluate robustness of topic segmentation methods. The *News* corpus contains 1180 segments while the *Reports* corpus only contains 140 segments.

The reference segmentation (i.e., ground truth) has been independently built by a user who was not involved in the design of a topic segmentation system. Since there is no consensus on the topic definition in the IR or NLP fields, it has been considered here that a topic change occurs for each report change. Despite this assumption being not always valid (in particular in the *News* corpus in which several successive reports may be considered as related to the same topic), it is relevant since it corresponds to an actual and well-defined applicative need.

Audio tracks of these two corpora have been automatically transcribed using the speech recognition system IRENE [13]. This system has been initially designed for transcribing radio broadcasts, including news, and is thus well-suited for our corpora. For these data, its Word Error Rate is about 20%, but this rate highly varies among the documents (e.g., anchor person speech vs. noisy outdoor speech). Transcriptions are finally part-of-speech tagged using TreeTagger[1], and only names, verbs, and adjectives are kept and stemmed.

5.2. Results

Recall (R), precision (P), and F1-measure (F1) are used as quality measures to evaluate our proposed method; we consider that a segment edge is correct as soon as it is located in the close neighbourhood (less than 10 seconds) of a reference frontier. We also indicate the WindowDiff measure (WD), which is usually preferred for evaluating segmentation systems [14]:

$$\text{WD(ref,hyp)} = \frac{1}{N-k}\sum_i |b(\text{ref}_i, \text{ref}_{i+k}) - b(\text{hyp}_i, \text{hyp}_{i+k})| > 0$$

where $b(x_i, x_j)$ is the number of boundaries between i^{th} and j^{th} utterances in the stream x, which contains N breath groups.

In order to show the relevance of our contribution, we compare the results obtained by our method to those produced by a baseline and by several existing systems on the same corpora. The baseline simply consists in dividing the document into as many (equal length) segments as there are segments in the reference. Concerning the existing systems, we use the DOTPLOT [3], TEXTTILING [5] and C99 [4] algorithms, as implemented by Choi [4] and adapted to French by Sitbon [7]. We also report the results, when available, of the system of Utiyama and Isahara [6] (as implemented in [8]) and the best results obtained from the system of [8]. Moreover, in order to assess the impact of vectorization similarity measure, we provide results obtained by our watershed approach using instead a simple TF-IDF/cosine distance. Thus the only difference is the way the gradient is computed (i.e., without vectorization), which can be here written:

$$\nabla(i) = cosine(\text{TF-IDF}(\mathcal{C}_{prev}(i-1)), \text{TF-IDF}(\mathcal{C}_{next}(i)))$$

[1] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger

For a fair comparison, it is worth noting that DOTPLOT, C99 and the baseline take as input the number of expected segments, while the other approaches do not.

Tables 1 and 2 show results obtained by all the systems on the two corpora. In both cases, we can observe that our system (Vectorization + Watershed) yields better results than existing systems, whatever the evaluation measure considered. It is also interesting to note that the watershed approach, even combined with a simple similarity measure as TF-IDF/cosine performs well. The superiority of Vectorization as a similarity measure is particularly observable on the *News* corpus, since this corpus contains very short segments, thus making the direct computation of the gradient as done in TF-IDF + Watershed approach unreliable. Moreover, in order to better understand the interest of using Watershed for topic segmentation, it is interesting to compare more deeply the approach introduced in this paper and TEXT-TILING. Indeed, the TEXT-TILING approach aims at finding topic breaks where lexical coherence between previous and next text blocks is linked to a significant local minimum. This approach is close to our ours but the coherence is computed based on a TF or TF-IDF/cosine measure, and the minima identified as those below a threshold based on the mean coherence.

Methods	P	R	F1	WD
Baseline	15.39	13.39	15.39	0.546
Utiyama [6]	-	-	59.44	-
Guinaudeau [8]	-	-	61.41	-
DOTPLOT [3]	36.42	36.42	36.42	0.4472
C99 [4]	50.25	50.25	50.25	0.3646
TEXTTILING [5]	41.96	35.96	38.73	0.313
TF-IDF + Watershed	44.17	41.97	43.04	0.3833
Vectorization + Watershed	**67.47**	**61.6**	**64.4**	**0.2269**

Table 1: Performance of topic segmentation systems on *News* corpus

Methods	P	R	F1	WD
Baseline	1.9	1.9	1.9	0.364
Utiyama [6]	-	-	51.09	-
Guinaudeau [8]	-	-	62.92	-
DOTPLOT [3]	49.49	49.49	49.49	0.2125
C99 [4]	57.42	57.42	57.42	0.1893
TEXTTILING [5]	25.96	21.27	23.38	0.3456
TF-IDF + Watershed	59.32	60.93	60.12	0.1844
Vectorization + Watershed	**77.38**	**69.65**	**73.31**	**0.1181**

Table 2: Performance of topic segmentation systems on *Reports* corpus

6. Conclusion

In this paper, we proposed a new topic segmentation algorithm based on a mathematical morphology tool, the watershed transform which is commonly used for image segmentation. Yet, the key component of our system is the gradient calculus which relies on the vectorization principle. This way to compute indirect similarity measure allows us to tackle the small-segment problem and the noisy data produced by the speech-to-text software. The experiments reported emphasize the interest of such an approach.

The image to text (or speech) analogy can be pushed further. Many improvements of the watershed and other approaches were proposed for image segmentation. We foresee their adaptation to our topic segmentation problems. In particular, hierarchical morphological segmentation schemes would be of great interest in our stream indexing framework in order to obtain a multiscale topic segmentation result.

7. References

[1] H. Christensen, B. Kolluru, Y. Gotoh, and S. Renals, "Maximum entropy segmentation of broadcast news," in *Proceedings of the 30th IEEE ICASSP*, 2005.

[2] M.-Y. Kan, J. L. Klavans, and K. R. McKeown, "Linear segmentation and segment significance," in *Proceedings of the 6th International Workshop of Very Large Corpora (WVLC-6)*, 1998.

[3] J. C. Reynar, "Topic segmentation: Algorithms and applications," Ph.D. dissertation, University of Pennsylvania, 2000.

[4] F. Y. Y. Choi, "Advances in domain independent linear text segmentation," in *Proceedings of the 1st metting of the North American Chapter of the Association for Computational Linguistics*, USA, 2000.

[5] M. Hearst, "Text-tiling: segmenting text into multi-paragraph subtopic passages," *Computational Linguistics*, vol. 23, no. 1, pp. 33–64, 1997.

[6] M. Utiyama and H. Isahara, "A statistical model for domain-independent text segmentation," in *Proceedings of the 9th conference of the ACL*, 2001.

[7] L. Sitbon and P. Bellot, "Adapting and comparing linear segmentation methods for french," in *7th International Conference RIAO, University of Avignon*, April 2004, pp. 623–637.

[8] C. Guinaudeau, G. Gravier, and P. Sébillot, "Improving asr-based topic segmentation of tv programs with confidence measures and semantic relations," in *Proc. Annual Intl. Speech Communication Association Conference (Interspeech)*, 2010.

[9] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583–598, 1991.

[10] V. Claveau, R. Tavenard, and L. Amsaleg, "Vectorisation des processus d'appariement document-requête," in *7e conférence en recherche d'informations et applications, CORIA'10*, Sousse, Tunisie, Mar. 2010, pp. 313–324.

[11] G. Salton, *A Theory of Indexing*, ser. Regional Conference Series in Applied Mathematics. Philadelphy: Society for Industrial and Applied Mathematics, 1975.

[12] I. Abraham, Y. Bartal, and O. Neiman, "Advances in metric embedding theory," in *Proc. of Symposium on Theory Of Computing*, Seattle, USA, 2006.

[13] S. Huet, G. Gravier, and P. Sébillot, "Morpho-syntactic post-processing with n-best lists for improved french automatic speech recognition," *Computer Speech and Language*, vol. 24, no. 4, pp. 663–684, October 2010.

[14] L. Pevzner and M. Hearst, "A critique and improvement of an evaluation metric for text segmentation," *Computational Linguistics*, 2002.