Unsupervised Learning with Term Clustering For Thematic Segmentation of Texts

Marc Caillet, Jean-Francois Pessiot, Massih-Reza Amini, and Patrick Gallinari

Computer Science Laboratory of Paris 6
8 rue du capitaine scott,
75015 Paris, France
{caillet,pessiot,amini,gallinari}@poleia.lip6.fr
http://www-connex.lip6.fr

Abstract. In this paper we introduce a machine learning approach for automatic text segmentation. Our text segmenter clusters text-segments containing similar concepts. It first discovers the different concepts present in a text, each concept being defined as a set of representative terms. After that the text is partitioned into coherent paragraphs using a clustering technique based on the Classification Maximum Likelihood approach. We evaluate the effectiveness of this technique on sets of concatenated paragraphs from two collections, the `7sectors` and the `20 Newsgroups` corpus, and compare it to a baseline text segmentation technique proposed by Salton et al.

1 Introduction

With the continuing growth of electronically available text resources, it is becoming more and more important to help users to access information and to develop easy to use information research tools. Thematic text segmentation may be helpful for that in the context of different information access tasks. It can be used together with conventional information search engines, and help users to quickly evaluate the relevance of documents or to navigate through a corpus. Text summarization is another task that can be improved by well-segmented text. For generic text summarization a summary can be generated by encapsulating pertinent subtopics which have been identified by a text segmenter.

Text segmentation has motivated a large number of publications and uncovers a variety of related but different tasks like the decomposition of long texts into topically coherent segments, or identification of topic changes in text streams like those used in Topic Detection and Tracking. Depending on the application it is used for, segmentation techniques differ according to the type of segment considered (concatenation of fixed text units e.g. sentences or paragraphs, word sequences when no text markup is available), the segment representation and the methods themselves. We focus here on thematic text segmentation which is the identification of the main themes of documents in a document collection. Following [15], we define text segments as paragraphs, a topic shift in this context is the boundary point between two distinct topically cohesive paragraphs. Our contribution is twofold, we first introduce a new method for representing paragraphs in a concept space, these concepts are automatically extracted from the collection and are identified by their set of representative words. This concept space offers several advantages compared e.g. to bag of words representations often used with statistical segmentation methods. In this space, segments which do not share the same words but discuss the same topic can be declared similar, it is thus well adapted to heterogeneous collections. Second text themes are discovered using a clustering technique which operates in this concept space, it is based on the Classification EM algorithm.

The remainder of this paper is as follows; we first make a brief review of work on text segmentation (section 2). In section 3 we present the framework of our model. Finally we present a series of experiments on a set of concatenated paragraphs from the 7sectors and the 20 Newsgroups data collections (section 4).

2 Related Work

Recently several innovative methods have been proposed for text segmentation. They can be roughly separated into two main families which respectively make use of statistical information extraction techniques and of lexical cohesion.

Many baseline statistical methods have been adapted for text segmentation. [13] proposed a contextual segmentation approach based on Local Context Analysis [19]. This technique uses term co-occurrence to map a query text onto semantically related words and phrases. [8] discusses a method for segmenting texts into multi-block subtopics called 'tiles' using the cosine similarity between segments. A similarity curve between adjacent blocks (defined as a fixed number of sentences ranging from 3 to 5) is computed and then smoothed to get rid of local extrema. Potential topic boundaries are then identified using the curve minima. [2] proposed a supervised learning algorithm which extracts text features correlated with the presence of boundaries in a labeled training text. In the context of Topic Detection and Tracking (TDT), [20] introduce hidden Markov models for text segmentation into homogeneous stories. In this approach, finding story boundaries is equivalent to finding topic transitions and segments are generated using unigram language models. [14] proposed a graphically motivated segmentation technique called dotplotting. This technique finds coherent regions using word repetition in the text. Statistical machine learning techniques have been recently introduced for the segmentation task. They allow a segmenting system to adapt to corpus characteristic in a fully automatic way. However, most of the approaches used for text segmentation rely on supervised learning and require the labeling of text boundaries for training which is usually performed manually [2]. This manual 'boundary-labeling' is unrealistic for large corpora.

Note that the thematic segmentation task we are dealing with is a very different task from the segmentation of text streams which has been extensively explored during the last years in the TDT context. Segmentation for TDT amounts at identifying the transition between two events which are usually related to very different topics. For thematic segmentation, the goal is to identify not only the frontier points but the different themes in a document collection from a set of topically coherent text segments. Transitions will usually be much less marked than for TDT.

[11] introduced the idea of lexical cohesion profile implemented into a semantic network. This technique computes the lexical cohesiveness between two words by activating the network node for one word and observing the activity value at an other word after a number of iterations in the network. The network is trained using a language specific knowledge source. More recently, [17] proposed a NLP approach based on the analysis of lexical cohesion within text for text segmentation. Their system first builds a structure made from a set of lexical chains and then searches for breaking points in this structure. [10] proposed an algorithm based on a dynamic programming technique to segment texts into variable levels of topical granularity. They first compute a matrix of similarity where each entry corresponds to the distance of a pair of sentences. They then use the anisotropic diffusion method [5] to enhance the lexical cohesion of sentences within a topical group. Finally the dynamic programming technique is adapted to segment texts. [9] proposed unsupervised techniques for word segmentation in Chinese information retrieval.

The work most similar to ours in this domain, is the well known study of [15]. The authors proposed to decompose texts into coherent thematic groups. The segmenting process begins at the paragraph level. They represent each paragraph using bag-of-words and infer cohesive multi-paragraph segments by comparing paragraphs using the cosine measure. This approach to thematic segmentation suffers from its relatively high complexity since in order to find text themes, each paragraph has to be compared with every other in the collection. This becomes prohibitive for large document collections.

Our text segmenter relies on unsupervised learning for paragraph clustering. Although the method used here is different, this is similar in spirit to the algorithm in [15] since it also relies on paragraph similarities. However, reasoning in the concept space allows for a considerable reduction of the algorithmic complexity. Another original aspect of our work compared to recent studies such as [9] which also focus on machine learning, is the use of unsupervised learning techniques for *thematic text segmentation*. Unsupervised learning has also been used for on line or off line `event detection` [1], a main task in TDT, but as said before, the goal, is rather different from thematic segmentation.

3 An Automatic Text segmenter

Our segmentation method is based on three successive operations. First concepts present in the collection are learned as a set of representative word-clusters. Second paragraphs are described in the space of these word clusters which provides a compact representation. Third, document themes are learned via the clustering of semantically related paragraphs. Finally, the system provides a set of segments taken to be the borders between two consecutive paragraphs . These three steps are detailed in the following sections.

3.1 Notations

For the remainder of the paper, we denote by $V = \{w_j\}_{j\in\{1,\dots,P\}}$ the set of P vocabulary terms, $D = \{x_i\}_{i\in\{1,\dots,n\}}$ the set of n paragraphs in the document collection and μ_k the coordinates of the k^{th} term cluster centroid c_k. As in [12], we denote by (j) the index of the centroid closest to term w_j. For example, $\mu_{(j)}$ denotes the closest centroid associated with w_j.

3.2 Concept learning

We define a concept as a cluster of terms formed according to the co-occurrence of words in paragraphs. A concept is then based on passage level evidence and on words co-occurring in local contexts. For discovering these term clusters, each term w in V is first characterized as an n-dimensional vector $\boldsymbol{w} =< n(w,i) >_{i\in\{1,\dots,n\}}$ representing the number of occurrence of w in each paragraph x_i.

Under this representation, term clustering is then performed using the `X-means` clustering algorithm [12] which is an extension of the well-known `K-means` algorithm [7] for which the number of clusters is estimated instead of being fixed by the user. Terms belonging to the same cluster will have similar representation in the paragraph space, which means that they frequently co-occur simultaneously in these segments. This algorithm searches the space of centroid locations in order to find the best partition of the input data. It starts from 2 partitions and then iteratively decides whether to split a cluster using the Bayesian Information Criterion (BIC). Splitting occurs when the information gain is increased compared to non splitting.

Formally, one assumes that each term w is generated independently from a mixture density

$$p(\boldsymbol{w}) = \sum_{k=1}^{K} \pi_k p(\boldsymbol{w} \mid y = k) \tag{1}$$

Where K is the number of term clusters and $\pi_k = p(y = k), k \in \{1, ..., K\}$ denote the mixing components. The $\boldsymbol{w}$ are vectors in the n-dimensional paragraph space. We have found that modeling $p(\boldsymbol{w} \mid y = k)$ using hyperspherical Gaussian densities with a common covariance matrix $\Sigma = \sigma^2.Id$ was a good compromise between efficiency and complexity. The maximum likelihood estimates of the centroids $\hat{\mu}_k$ and the variance $\hat{\sigma}^2$ are:

$$\hat{\mu}_k = \frac{1}{|c_k|} \sum_{\boldsymbol{w}_j \in c_k} \boldsymbol{w}_j$$

$$\hat{\sigma}^2 = \frac{1}{P - K} \sum_j \|\boldsymbol{w}_j - \mu_{(j)}\|^2$$

Therefore the maximum log-likelihood of cluster c_k obtained using model Θ_m is:

$$\hat{l}_m(c_k) = \sum_{\boldsymbol{w}_j \in c_k} \left(\frac{1}{\sqrt{2\pi}\hat{\sigma}} - \frac{1}{2\hat{\sigma}^2} \parallel \boldsymbol{w}_j - \mu_k \parallel^2 + \log \pi_k \right) \tag{2}$$

Where, $\| \,.\, \|$ is the Euclidean norm. The best model in the Θ_m family is the one which maximizes BIC criterion defined as:

$$BIC(\Theta_m) = \sum_k \hat{l}_m(c_k) - \frac{p_m}{2} \log P \tag{3}$$

where, p_m is the number of parameters in Θ_m, i.e. the means coordinates and the covariance matrix in our case.

`X-means` begins from two initial clusters obtained by `K-means` with `K`=2 onto the entire vocabulary and a maximum number of iterations T. It then splits iteratively each existing cluster with `2-means` and replaces the cluster by its two children if the splitting increases the BIC criterion. The algorithm thus searches the entire vocabulary for the best region to split.

Table 1, shows some term clusters found for both data sets, it can be seen from this example that each cluster can be associated a general concept. In the following section, we present a paragraph characterization using the clusters obtained with `X-means`. This new feature representation leads to an efficient paragraph representation.

Table 1. An example of term clusters found with `X-means` in `7-sectors` (top) and `20 Newsgroups` (down) data collections.

Cluster i**:** `video feature graphic format info print adobe acrobat reader connection browse html cooper valve animation modem printing`
Cluster j**:** `heart pain trial lung blood condition effect tissue body injury inflammatory license stroke channel rejection therapeutic swelling brain transplantation inhibitor organ protein receptor enzyme attack activation calcium neuron glutamate nerve cascade complement`

Cluster k**:** `wrong christian words christians truth meaning paul john bible word faith fact reason men`
Cluster l**:** `suicide suicides deaths stat selfdefense risks statstic guns` accidents homicides nejm

3.3 Dimensionality reduction

We will represent text segments, here paragraphs, in the space of the learned concepts. To do that, paragraph x_i will be represented as a vector

$$\boldsymbol{x_i} =< \bar{n}(c,i) >_{c\in\{1,\ldots,|C|\}} \tag{4}$$

where feature $\bar{n}(c,i)$ represents the number of occurrences of terms from cluster c in the paragraph. The characteristics of a paragraph in this representation are related to the degree of representation of each concept in this paragraph. This is the concept space where paragraphs will be compared in the third step. One side effect of this representation is to decrease the complexity of this third step. However, the main idea is that similarities in this space are more significant than in the vocabulary term space which is used for example in [15].

We will introduce a refinement into this scheme by allowing 'polysemic' terms. We consider that a term is polysemic if it is related to different learned concepts. Such a term appears frequently with set of words characteristic of these different learned concepts. Practically, we will consider that these terms do appear at the frontier of different clusters. We thus define the set of concepts for each term in the vocabulary as

$$C_{polysemic}(w) = \{c \in C \mid \|w - \mu_c\| \leq d_w + \epsilon\} \tag{5}$$

d_w represents the distance of w to its nearest centroid in C and ϵ is a preset parameter. Term w is considered 'polysemic' in this sense if $|C_{polysemic}(w)| > 1$. Such terms deserve a special treatment. For each polysemic w in paragraph x_i, we assign w to its most frequent cluster in x_i instead of its most frequent cluster in the collection. w will score for this *most frequent paragraph cluster* instead of its *most frequent collection cluster*. Each term may then have a sense specified by its local context. This allows to disambiguate term senses by choosing its most frequent term cluster in $C_{polysemic}$ in a paragraph.

Using this feature characterization, we present, in the following section, a clustering technique based on the Classification Maximum Likelihood approach for paragraph clustering and text segmentation.

3.4 Paragraph Clustering and document segmentation

The classification maximum likelihood (CML) approach [18] is a general framework which encompasses many clustering algorithms for unsupervised learning [4]. Let us introduce the classification likelihood L_{CML} :

$$L_{CML} = \sum_{i=1}^{n} \sum_{k=1}^{\Omega} t_{ki} \ \log p(\boldsymbol{x}_i, y = k) \tag{6}$$

where n is the number of paragraphs, Ω is the number of paragraph clusters and t_{ki} is 1 if x_i belongs to cluster k and 0 otherwise. The n samples (paragraphs) are supposed to be generated via a mixture density :

$$p(\boldsymbol{x}) = \sum_{k=1}^{\Omega} \pi_k \ p(\boldsymbol{x} \mid y = k) \tag{7}$$

The parameters of the mixture are estimated by maximizing the CML criterion. Note that this is different from the classical mixture maximum likelihood (MML) approach which optimizes the data likelihood :

$$L_{MML} = \sum_{i=1}^{n} \log \sum_{k=1}^{\Omega} \pi_k \ p(\boldsymbol{x}_i \mid y = k) \tag{8}$$

For MML the goal is to model the data distribution components which can then be used to cluster the data, whereas the CML approach directly aims at clustering the data which is exactly our goal here. For CML the mixture indicator t_{ki} for a given example x_i is treated as an unknown parameter of the model and has to be estimated together with these parameters. In this case the t indicators correspond to a hard decision on the mixture component density.

The classification EM algorithm (CEM) is an iterative technique [4] similar to classical EM except for an additional `C`-step where each $\boldsymbol{x}_i$ is assigned to one and only one component of the mixture[1]. The algorithm is sketched below. We suppose that paragraphs are generated independently from a mixture density and that each mixture component obeys a naive Bayes model. The parameters of this model are the set of class priors $\pi_k = p(y = k)$ and binomials p_{ck} which denote the probability of apparition of occurrences of terms from the cluster c in a paragraph (4). Under these assumptions, $p(\boldsymbol{x} \mid y = k) \equiv \prod_{c=1}^{|C|} p_{ck}^{\bar{n}(c,x)}$.

Algorithm 1 CEM

```
*Initialization*: start from an initial partition $\Pi^{(0)}$
$j^{th}$ iteration, $j \geq 0$:
  – E-step: Estimate the posterior class probability that $x_i$ belongs to $\Pi_k$ $(i = 1, ..., n; k = 1, ..., c)$:
      $$E[t_{ki}^{(j)} \mid x_i; \Pi^{(j)}] = \frac{p(y^{(j)} = k)\ p(\boldsymbol{x}_i \mid y^{(j)} = k)}{\sum_{k=1}^{\Omega} p(y^{(j)} = k)\ p(\boldsymbol{x}_i \mid y^{(j)} = k)} \qquad (9)$$
  – C-step: Assign each $\boldsymbol{x}_i$ to the cluster $\Pi_k^{(j+1)}$ with maximal posterior probability according to $E[t \mid x]$.
  – M-step: Estimate the new parameters for the model which maximize $L_{CML}$.
```

Differentiating (6) with respect to π_k and p_{ck} and introducing Lagrange multipliers to enforce the constraints $\sum_k \pi_k = 1$ and $\forall k, \sum_c p_{ck} = 1$, we get the maximum likelihood estimates of π_k and p_{ck} :

$$\pi_k = \frac{\sum_{i=1}^{n} t_{ki} + 1}{n + \Omega} \quad \text{and} \quad p_{ck} = \frac{\sum_{i=1}^{n} t_{ki}\ \bar{n}(c, x_i) + 1}{|x_i| + |C|} \qquad (10)$$

Where, $|x_i| = \sum_{c=1}^{|C|} \bar{n}(c, x_i), \forall i$. Note that the discriminant function in this case is linear in the observations.

To segment a text, candidate points of segmentation are identified as those corresponding to locations between text blocks. Each text block is constituted as a set of different paragraphs belonging to a given cluster found in this step.

[1] The `K-means` algorithm is a particular instance of the CEM algorithm in which data are supposed to be drawn from hyper spherical gaussian distributions

4 Experiments

The corpus For our experiments, we used two data collections from the CMU WebKB project[2]. The first data set is the 20 Newsgroups[3] database consisting in 20017 Usenet articles taken from twenty newsgroups at random (about 1000 documents per newsgroup) and labeled by their newsgroup name. In our experiments, 4% of documents in the data set belonging to more than one newsgroup were discarded. Table 2 (up), gives the list of these news groups from which the documents were chosen.

The second data set is the 7sectors[4] database consisting in 3417 *html* articles partitioned in hierarchical order. We labeled each document in this collection with its initial parent class label, namely basic, energy, financial, health, transportation, technology and utilities. Table 2(down), summarizes the characteristics of this collection.

Table 2. A list of the 20 news groups (up) and characteristics of the 7sectors dataset (down).

20 Newsgroups dataset				
comp.graphics	rec.autos	sci.crypt	talk.politics.misc	talk.religion.misc
comp.os.ms-windows.misc	rec.motorcycles	sci.electronics	talk.politics.guns	alt.atheism
comp.sys.ibm.pc.hardware	rec.sport.baseball	sci.med	talk.politics.mideast	soc.religion.christian
comp.sys.mac.hardware	rec.sport.hockey	sci.space		
comp.windows.x	misc.forsale			

7sectors dataset		
Class	size	proportion %
basic	714	20.9
energy	265	7.7
financial	697	20.4
health	310	9.1
technology	823	24.1
transport	383	11.2
utilities	225	6.6

After removing all *html* tags and *news headrs* of respectively 7sectors and 20 newsgroups documents, our pre-processing included lowering upper case characters and ignoring non alphanumeric characters. We also removed stop words and words that occurred in less than 3 documents, ending up with a vocabulary of 16252 terms for 7sectors and 28101 for 20 Newsgroups.

Figure 1 shows the document length in words for both datasets. The average word length is about 177 terms or 17 sentences for 7sectors and 81 terms or 11 sentences for 20 Newsgroups. In light of these observations, it is not unrealistic to follow [17] by considering a document from these datasets as a paragraph for our experiments.

From these two collections, where each document is clearly associated to one theme, we build two new collections for the evaluation of segmentation algorithms, by concatenating at random

[2] http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-11/www/wwkb/

[3] http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/news20.tar.gz

[4] http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/bootstrappingIE/7sectors.tar.gz

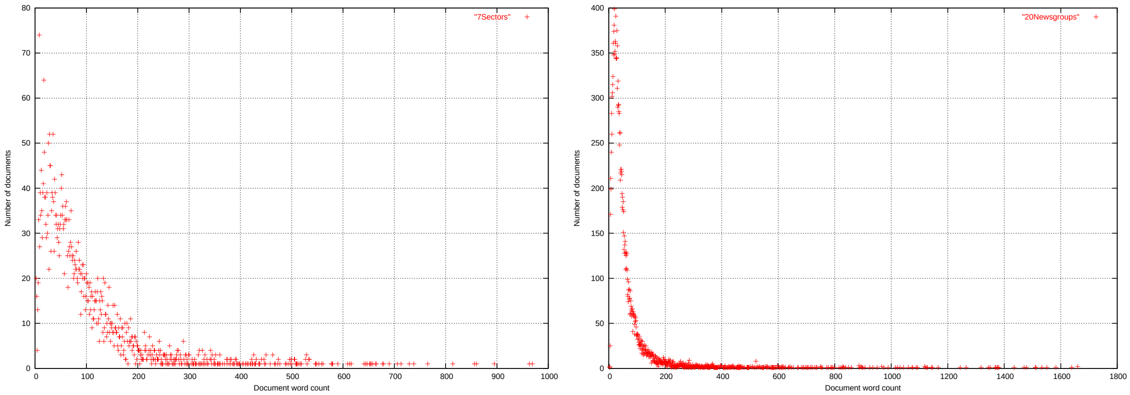

Fig. 1. Distribution of document word length for `7Sectors` (left) and `20 Newsgroups` (right) data collections .

distinct documents from respectively the `7sectors` and the `20 Newsgroups` datasets. In each text collection, we define a text block as a set of concatenated paragraphs. A segment boundary then corresponds to a topic shift between two consecutive text blocks. This collection has been built for evaluation purpose. It is probably not representative of document collections for which thematic segmentation has been proposed, but it allows to define explicitly segments as the joining point between two text blocks. A topic or a theme for each text block is a class label from the database the block is originated. In order to find the same number of clusters than class labels in each data collection, we set the number of paragraph clusters, Ω, to respectively 7 for `7sectors` and 20 for `20 Newsgroups`. Once again, in a real situation, we do not know the number of themes present in the collection, so that the number of clusters has to be estimated from the data. However it then becomes extremely difficult to objectively compare document partitions with different number of classes. This is why we chose this direct approach to the evaluation problem.

Table 3 contains information about the number of concepts found for both data sets using the `X-means` algorithm. Representing paragraphs in the learned concept space rather than the original vocabulary space leads to a dimensionality reduction equal to approximately 1/79 in both test collections. Figure 2, plots the distribution of concepts within paragraphs for both test collections. These distributions are narrowly lognormals with an expected mean around 40 concepts for the `7sectors` and 25 concepts for the `20 Newsgroups`.

Table 3. Number of concepts found for `7Sectors` and `20 Newsgroups` data collections.

Data set	Vocabulary size	# of concepts
`7sectors`	16252	217
`20 Newsgroups`	28101	361

For these test sets, we ran 3 algorithms - our text segmenter using the above concept space representation for paragraphs (COS with CEM), the CEM algorithm with a simple bag-of-words characterization of paragraphs (BOW with CEM) and the Salton et al. text segmenter [15]. The BOW with CEM is a clustering system which bears similarity with those used for event detection.

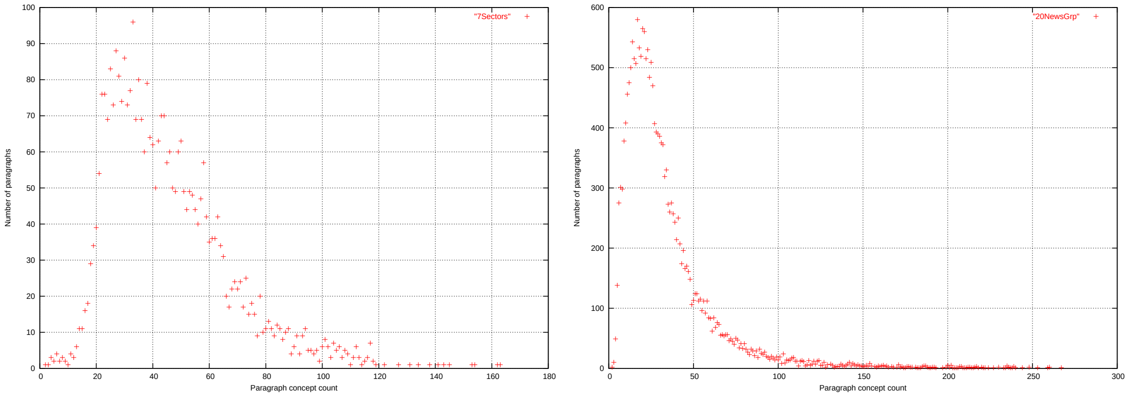

Fig. 2. Distribution of paragraph concept count for `7Sectors` (left) and `20 Newsgroups` (right) data collections .

The evaluation measure

For evaluation measures, we used the micro-averaged precision and recall. For the estimation of these measures, we followed [16]. We first assigned all paragraphs in a given cluster to the most dominant class label in that cluster. For each class, we estimate the following quantities:

$\alpha(c)$: the number of paragraphs correctly assigned to c,
$\beta(c)$: the number of paragraphs incorrectly assigned to c,
$\gamma(c)$: the number of paragraphs incorrectly not assigned to c.

For a given class the precision and recall measures are defined as $\text{Precision}(c) = \frac{\alpha(c)}{\alpha(c)+\beta(c)}$ and $\text{Recall}(c) = \frac{\alpha(c)}{\alpha(c)+\gamma(c)}$. The micro-averaged of these measures are defined as :

$$\text{Precision} = \frac{\sum_{\text{all } c} \alpha(c)}{\sum_{\text{all } c} \alpha(c) + \beta(c)} \qquad \text{Recall} = \frac{\sum_{\text{all } c} \alpha(c)}{\sum_{\text{all } c} \alpha(c) + \gamma(c)}$$

It is easy to see that $\sum_c \alpha(c) + \beta(c) = \sum_c \alpha(c) + \gamma(c)$ and then on average, Precision and Recall are equal.

Results

In tables 4 and 5, we present the micro average precision results respectively for the `7sectors` and the `20 Newsgroups` data collections. Our text segmenter system (COS with CEM) is over the other two systems in both test sets. Salton et al. system has found one thematic class by gathering roughly all paragraphs in one thematic cluster. We note that, some clusters have been merged into a single one. This effect is emphasized in the `7sectors` data set which is more heterogeneous than the `20 Newsgroups` collection. We have carefully checked our implementation of this algorithm and concluded that this is inherent to the method which is based on the detection of word regularities and therefore is very sensible to the large vocabularies present in heterogeneous collections. By comparing the segmentation models based on CEM it comes that the COS system leads not only to a more concise representation of paragraphs but also characterizes paragraphs in a more efficient manner than the bag-of-words representation.

In Figure 3, we present the precision-recall curves of the different systems for the `7sectors` test set. These curves confirm the ranking of the different systems.

Table 4. Micro average Precision results for different text segmenters over the `7sectors` dataset, α, β and γ respectively correspond to the number of paragraphs correctly assigned to c, incorrectly assigned to c and incorrectly not assigned to c.

`7sectors dataset`									
	COS with CEM			BOW with CEM			Salton et al.		
	$\alpha(c)$	$\beta(c)$	$\gamma(c)$	$\alpha(c)$	$\beta(c)$	$\gamma(c)$	$\alpha(c)$	$\beta(c)$	$\gamma(c)$
basic	507	789	207	461	977	253	0	0	714
energy	0	0	265	0	0	265	0	0	265
finan.	357	125	340	400	350	297	2	0	695
health	0	0	310	0	0	0	0	0	310
technol.	613	624	210	583	646	240	823	2591	0
transport.	199	203	184	0	0	383	1	0	382
utilities	0	0	225	0	0	225	0	0	225
Average Precision	**49.05**			**42.25**			**24.17**		

Table 5. Micro average Precision results for different text segmenters over the `20 Newsgroups` datasets.

`20 News groups dataset`									
	COS with CEM			BOW with CEM			Salton et al.		
	$\alpha(c)$	$\beta(c)$	$\gamma(c)$	$\alpha(c)$	$\beta(c)$	$\gamma(c)$	$\alpha(c)$	$\beta(c)$	$\gamma(c)$
alt.atheism	553	998	418	0	0	971	0	0	971
comp.graphics	220	131	753	119	95	854	0	0	973
comp.os.ms.win.	0	0	962	0	0	962	0	0	962
comp.sys.ibm.pc.hw	767	2276	222	940	4100	49	0	0	989
comp.sys.mac.hw	0	0	975	0	0	975	0	0	975
comp.windows.x	654	488	314	587	121	381	0	0	968
misc.forsale	22	10	942	0	0	964	1	0	963
rec.autos	838	4822	126	829	1865	135	1	0	963
rec.moto	0	0	967	23	136	944	2	0	965
rec.sport.bsball	0	0	955	0	0	955	0	0	955
rec.sport.hockey	807	636	167	921	927	53	0	0	974
sci.crypt	602	185	387	210	274	779	989	18455	0
sci.electronics	241	250	735	0	0	976	0	0	976
sci.med	390	104	589	755	462	224	8	0	971
sci.space	380	153	601	195	275	786	3	0	978
soc.religion.chrs	585	452	400	900	2460	85	0	0	985
talk.politics.guns	655	990	324	755	1593	224	2	0	977
talk.politics.mideast	670	358	289	329	346	630	0	0	959
talk.politics.misc	46	97	934	10	58	970	1	0	979
talk.religion.misc	22	62	952	38	140	935	2	0	972
Average Precision	**38.28**			**33.97**			**5.18**		

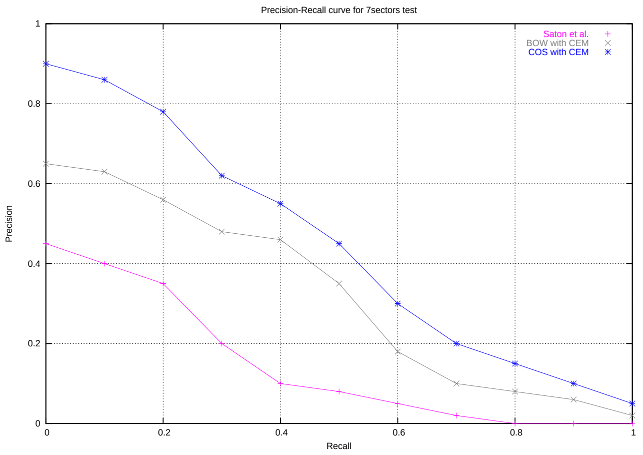

Fig. 3. Precision-Recall curves for different text segmentation methods over the `7sectors` dataset.

5 Conclusion

We have proposed a new general unsupervised approach for training text segmenters based on paragraph extraction and performed an evaluation on `7sectors` and `20 Newsgroups` data collections. In this approach, paragraphs are encoded into a learned concept space allowing an important dimensionality reduction. Our method has been compared to Salton et al.'s text segmenter [15] and to an unsupervised text segmenter based on the CEM algorithm but using a classical bag-of-words representation as it is classically used in `event detection`. The proposed algorithm outperforms these two systems.
In future work, we plan to evaluate our systems on large and heterogeneous data sets.

References

1. Allan. J., Jin, H., Rajman. M., Wayne, C., Gildea, D., Lavrenko., V., Hoberman., R., Caputo, D. : Topic-based Novelty Detection: Final Report. In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop. (1999)
2. Beeferman, D., Berger, A., Lafferty, J. : Statistical Models for Text Segmentation. in Machine Learning, (34):177-210. Special Issue on Natural Language Learning (C. Cardie and R. Mooney, eds). (1999)
3. Carbonell, J., Yang, Y., Lafferty, J., Brown, R., Pierce, T., Liu, X. : CMU Report on TDT2: Segmentation Detection and Tracking Proceedings of the DARPA Broadcast News Workshop, pp.117-120. (1999)
4. Celeux G., Govaert, G. : A Classification EM algorithm for clustering and two stochastic versions. Computational Statistics and Data Analysis, 14(3):315–332. (1992)
5. Clarenz, U., Diewald, U., Rumpf, M. : Anisotropic Geometric Diffusion in Surface Processing. Proceedings Visualization 2000, pp. 397-405 (2000)
6. Dempster, A., Laird N., Rubin, D. : Maximum likelihood from incomplete data via the EM algorithm. Journal of Royal Statistical Society, Series B, 39(1):1–38. (1977)
7. Duda, R.O., Hart, P.E. : Pattern Recognition and Scene Analysis. J. Wiley and Sons, New-York. (1973)
8. Hearst, M. : TextTiling: a Quantitative Approach to Discourse Segmentation. Technical Report, UCB:S2K-93-24, pp.33-64. (1993)
9. Huang, X., Peng, F., Schuurmans, D., Cercone, N., Robertson. S.E. : Applying Machine Learning to Text Segmentation for Information Retrieval. Information Retrieval, Vol. 6, N. 3/4, pp.333-362. (2003)

10. Ji, X., Zha, H., : Domain-independent Text Segmentation Using Anisotropic Diffusion and Dynamic Programming. in Proceedings of the 26^{th} Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.322-329. (2003)
11. Kozima, H. : Text Segmentation Based on Similarity Between Words. in Proceedings of ACL-93, pp.286-288. (1993)
12. Pelleg, D., Moore, A. : X-means: Extending K-means with Efficient Estimation of the Number of Clusters. in Proceedings of 17th International Conf. on Machine Learning. pp.727-734. (2000)
13. Ponte, J.M., Croft, W.B. : Text Segmentation by Topic. in Proceedings of the First European Conference on research and advanced technology for digital libraries. (1997)
14. Reynar, J.C. : An Automatic Method of Finding Topic Boundaries. in Proceedings of ACL-94, pp.331-333. (1994)
15. Salton, G., Singhal, A., Buckley, C., Mitra, M. : Automatic Text Decomposition Using Text Segments and Text Themes. in Proceedings of the Seventh ACM Conference on Hypertext, pp.53-65. Washington D.C (1996)
16. Slonim, N., Friedman, N., Tishby N. : Unsupervised Document Classification using Sequential Informaiton Maximization. in Proceedings of the Twenty-Fifth Annual International ACM SIGIR Conference pp.129-136. Tampere, Finland (2002)
17. Stokes, N., Carthy, J., Smeaton, A.F. : Segmenting Broadcast News Streams Using Lexical Chains. in Proceedings of 1st Starting AI Researchers Symposium (STAIRS 2002), volume 1, pp.145-154. (2002)
18. Symons, M. J. Clustering criteria and multivariate normal mixture. Biometrics, 37(1):35–43. (1981).
19. Xu, J., Croft W.B. : Query Expansion Using Local and Global Document Analysis. in Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.4-11. Zurich, Switzerland (1996)
20. Yamron, J.P., Carp, I., Gillick, L., Lowe, S., van Mulbregt, P. : Text Segmentation and Topic Tracking on Broadcast News via a Hidden Markov Model Approach. in ICSLP98, Volume VI, pp.2519-2522. (1998)
21. Yang, Y., Carbonell, J.G., Brown, R.D., Pierce, T., Archibald, B.T., Liu X. : Learning Approaches for Detecting and Tracking News Events. In Intelligent Information Retrieval, July 1999, pp. 32-43. (1999)