

---

# Apprentissage non-supervisé pour la segmentation automatique de textes

Jean-François Pessiot, Marc Caillet, Massih-Reza Amini, Patrick Gallinari

Laboratoire d'Informatique de Paris 6  
8 rue du Capitaine Scott  
75015 Paris, France  
{pessiot, caillet, amini, gallinari}@poleia.lip6.fr  
[http : //www - connex . lip6 . fr](http://www-connex.lip6.fr)

---

*RÉSUMÉ.* Nous proposons dans cet article une approche basée sur des techniques d'apprentissage pour la segmentation automatique de texte. Nous considérons un paragraphe comme l'entité textuelle de base. Notre système découvre d'abord différents concepts présents dans un texte, chaque concept étant défini par un ensemble représentatif de mots. Le texte est ensuite segmenté suivant des paragraphes en utilisant une technique de partitionnement basée sur la vraisemblance classifiante. Nous évaluons l'efficacité de cette technique sur un ensemble concaténé de paragraphes de la collection *7sectors* et nous la comparons à une technique de référence proposée par Salton et al.

*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 hard clustering technique based on the Classification Maximum Likelihood approach. We evaluate the effectiveness of this technique on a set of concatenated paragraphs from the *7sectors* data collection and compare it to a well-established text segmentation technique proposed by Salton et al.

*MOTS-CLÉS :* Segmentation de texte, Apprentissage non-supervisé, Partition de mots, Vraisemblance classifiante.

*KEYWORDS:* Text segmentation, Unsupervised learning, Term clustering, Classification Maximum Likelihood.

---

## 1. Introduction

Avec le développement de systèmes de communication électronique de plus en plus performants et l'accroissement de la quantité d'information disponible en ligne, il devient de plus en plus important d'aider les utilisateurs à accéder plus rapidement à l'information recherchée et à développer de nouveaux outils de recherche d'information. A cette fin, la segmentation thématique de texte pourrait être considérée comme un outil d'aide pour différentes tâches d'accès à l'information. Elle peut être employée conjointement à l'utilisation de moteurs de recherche conventionnels pour aider les utilisateurs à évaluer rapidement la pertinence des documents retournés en réponse à une requête, ou encore pour faciliter la navigation à travers un corpus conséquent. Le résumé de texte est une autre tâche dont le résultat peut être amélioré par la segmentation thématique. Un résumé peut ainsi être généré à partir de différentes thématiques pertinentes identifiées par un système de segmentation.

La plupart des approches pour la segmentation proposées reposent sur des méthodes statistiques ou linguistiques. Les systèmes linguistiques sont souvent construits spécifiquement pour un corpus donné et dans ce cas difficilement transposables à une autre collection. Les méthodes statistiques, quant à elles, reposent le plus souvent sur l'approche de 'sac de mots' (bag of words) pour représenter une unité textuelle (paragraphe ou phrase). Elles échouent souvent sur des corpus hétérogènes. L'ensemble des méthodes est d'autre part souvent peu apte à traiter des corpus de grande taille. du fait de leur combinatoire.

Récemment des méthodes basées sur l'apprentissage ont été proposées pour la segmentation. L'argument en faveur de ces méthodes est d'être capable de s'adapter à des conditions opérationnelles bien plus diverses et en particulier de s'adapter à différents types de corpus. Toutefois, les approches actuelles sont basées sur des méthodes supervisées (e.g. [BEE 97]). Elles nécessitent, à ce titre, l'étiquetage manuel de textes au niveau phrase ou paragraphe et, par conséquent sont également limitées dans leur utilisation sur des corpus hétérogènes et de grande taille.

L'apport de notre travail est d'une part de proposer une représentation des segments de texte dans un espace de dimension réduite appelé *espace des concepts* qui est plus riche que la représentation sac de mots, et d'autre part de proposer un système de segmentation basé sur l'apprentissage non-supervisé, où il n'est plus nécessaire d'étiqueter manuellement un corpus pour apprendre. Les résultats de cette méthode sont comparés à ceux obtenus par une méthode de référence développée par [SAL 96] sur le corpus de *7sectors*.

La suite de l'article est organisée comme suit : nous passons en revue les différentes approches proposées pour la segmentation de texte dans la section 2. Nous présentons dans la section 3 notre système à base d'apprentissage non-supervisé pour cette tâche. Dans la section 4, nous évaluons le système sur un corpus obtenu en concaténant les documents de la collection *7sectors* et nous concluons en section 5.

## 2. Travaux connexes

Nous allons tout d'abord présenter quelques algorithmes classiques pour la segmentation thématique de textes. Ces algorithmes cherchent à segmenter un texte suivant différents types de segments : *sémantiques*, de *discours* ou *contextuels*. Nous distinguerons trois familles de méthodes. La première se concentre sur la cohésion lexicale, la seconde utilise des techniques statistiques pour calculer une mesure de similarité entre segments successifs et la dernière utilise des méthodes issues de la communauté "apprentissage numérique". Dans toutes ces méthodes, l'unité de base utilisée est la phrase, i.e. un segment thématique sera composé d'une suite de phrases, sauf pour la méthode proposée par [SAL 96] qui utilise comme unité le paragraphe. Ce choix se comprend par le fait que le paragraphe est un élément

textuel de plus grande taille qu'une unité plus restreinte comme un groupe de mots et une phrase, et il peut par ce fait mieux exprimer l'idée d'un thème.

## 2.1. Cohésion Lexicale

### 2.1.1. Cohésion lexicale simple

[WAL 91] étudie la répétition de termes dans les phrases comme indicateur de cohésion. La valeur de cet indicateur est utilisée pour trouver les points de segmentation. Le facteur de répétition a une importance variable suivant les différents types de textes étudiés. Il existe des textes (comme les articles scientifiques) ayant un vocabulaire spécifique et où le nombre d'homonymes est souvent réduit. À l'opposé, il y a des textes où le nombre de termes, exprimant une même notion, est souvent élevé et dans ce cas la répétition de termes n'est pas un bon indicateur de cohésion.

[KAN 98] découpent linéairement un texte en étudiant les chaînes lexicales présentes dans le texte cible. Ces chaînes relient les occurrences d'un terme dans les phrases des documents à segmenter. Une chaîne est rompue si le nombre de phrases séparant deux occurrences est trop important. Ce nombre de phrases dépend de la catégorie syntaxique du terme considéré. Cette catégorie est utilisée conjointement avec la longueur des chaînes pour affecter un poids à ces dernières. Un score est ensuite donné à chaque paragraphe en fonction des poids et des caractéristiques des chaînes qui le traversent. Des marques de segmentation sont apposées au début des paragraphes ayant les scores maximaux.

### 2.1.2. Cohésion lexicale étendue

Dans le cas où un concept serait exprimé par des termes différents, l'analyse de la répétition des termes n'est plus suffisante pour détecter les marques de transition thématique. [KOZ 93] propose de calculer un coefficient de cohésion lexicale (*Lexical Cohesion Profile*) pour chaque segment du texte en fonction de la ressemblance des mots présents en utilisant un réseau de collocations. Pour chaque position d'une fenêtre dans le texte, il calcule une valeur de cohésion de cette fenêtre en fonction du poids des mots dans la fenêtre et du poids des mots du réseau de collocations contenant au moins deux termes communs à ceux de la fenêtre. Le déplacement de la fenêtre permet de distinguer les zones de forte et de faible cohésion.

[PON 97] présentent une approche différente : ils utilisent la phrase comme unité textuelle de base. En commençant par une phrase, ils l'enrichissent d'abord en utilisant la méthode LCA (*Local Context Analysis*) qui est une méthode d'enrichissement de requête basée sur le retour utilisateur. En utilisant des techniques de programmation dynamique, ils déterminent ensuite les phrases à droite de la phrase enrichie qui lui sont sémantiquement proches. Plus récemment, [STO 02] ont proposé une approche TALN pour l'analyse de la cohésion lexicale des mots d'un texte, dans le cadre de la segmentation, au niveau de la phrase, des brèves d'information concaténées. Leur système crée tout d'abord une séquence de chaînes lexicales, ensembles de mots du texte reliés entre eux par des relations sémantiques telles que définies par le réseau sémantique WordNet [MIL 90]. Il détermine ensuite les points de rupture entre les phrases, créant ainsi les segments.

## 2.2. Approche statistique

[HEA 97] présente une méthode de segmentation de textes en sous-thèmes nommée *Text tiling*. Pour ce faire, elle propose de comparer des paires de blocs adjacents du texte, en faisant l'hypothèse que deux blocs sont similaires s'ils traitent du même sujet (la similarité étant évaluée par un calcul de cosinus sur les blocs représentés comme des sacs de mots). Malgré de bonnes performances sur des textes simples et homogènes, cette approche est peu adaptée à des textes hétérogènes ou présentant

une grande variabilité. Les travaux développés par [SAL 96] sont ceux qui se rapprochent le plus de la méthode que nous présentons ici, dans le sens où des segments non adjacents peuvent être regroupés au sein d'une même thématique. La segmentation opère au niveau du paragraphe comme suit : chaque paragraphe est représenté sous la forme d'un vecteur de fréquences de mots, et les thématiques sont construites en comparant les paragraphes entre eux avec la mesure cosinus. Cette approche souffre d'une complexité importante car, pour déterminer les différentes thématiques, chaque paragraphe doit être comparé à tous les autres, ce qui induit un coût prohibitif et rend la tâche irréalisable dans le cas du traitement de grandes collections de documents. Nous proposons dans cet article de traiter de telles collections, d'une part grâce à la complexité réduite de notre algorithme de segmentation, d'autre part grâce à l'introduction de concepts sémantiques dans l'espace desquels sont représentés les paragraphes, réduisant ainsi considérablement la dimension de l'espace de représentation.

### **2.3. Méthodes basées sur l'apprentissage**

Depuis le milieu des années 90 de nouvelles méthodes de segmentation ont été présentées reposant sur l'utilisation de classifieurs ou d'algorithmes ad-hoc.

[LIT 95, BEL 01] utilisent des arbres de décision pour la segmentation. [LIT 95] s'appuient sur des caractéristiques linguistiques déduites de corpus oraux. Pour trouver les phrases contenant la réponse à une requête posée en langage naturel, [BEL 01] proposent de construire un arbre de décision par question. [BEE 97] construisent un modèle probabiliste exponentiel qui à chaque phrase fait correspondre la probabilité qu'il y ait une frontière entre cette phrase et la phrase suivante. La distribution de probabilité est choisie en construisant de façon incrémentale un modèle log-linéaire. Dans [BIG 98], un certain nombre de thématiques sont apprises sur une base d'apprentissage. Les auteurs emploient alors un modèle de langage pour étiqueter chaque paragraphe d'un texte suivant sa thématique de plus forte probabilité. Lorsque la valeur de la probabilité du meilleur thème décroît, une frontière thématique possible est détectée. La sélection définitive des frontières est effectuée suivant une méthode de programmation dynamique. [YAM 98] présentent une méthode de segmentation basée sur les Modèles de Markov Cachés (MMC). Les états cachés du MMC représentent des thèmes prédéfinis et les observations sont des mots ou des phrases. Ils divisent d'abord le texte en régions sémantiquement proche en utilisant l'algorithme des K-moyennes en utilisant la distance de Kullback. Les transitions entre les états sont ensuite déterminées à la main par rapport à un petit nombre de thématiques analysées par des experts.

Toutes ces méthodes sont supervisées exceptée celle de [BEL 01] qui concerne la segmentation par rapport à une requête et non pas la segmentation d'un texte en ses différentes thématiques.

## **3. Un système automatique de segmentation**

Notre méthode de segmentation considère comme unité de base le paragraphe, elle comporte trois étapes successives. On apprend tout d'abord les concepts, chacun étant défini comme un ensemble représentatif de mots. Dans une deuxième étape, on caractérise les paragraphes dans l'espace de ces concepts. Cette étape permet de représenter les paragraphes d'une manière concise en réduisant considérablement la dimensionnalité du problème par rapport à la représentation sac de mots. On trouve finalement les différentes thématiques présentes dans la collection en regroupant les paragraphes "sémantiquement" proches au sens de ces concepts. Notre modèle fournit une collection de segments étiquetés suivant leur concept. Les trois étapes sont détaillées dans les sections suivantes.

### 3.1. Notation

Par la suite, on notera par  $V = \{w_j\}_{j \in \{1, \dots, P\}}$  l'ensemble des  $P$  mots du vocabulaire,  $D = \{x_i\}_{i \in \{1, \dots, n\}}$  l'ensemble des  $n$  paragraphes dans la collection et  $\mu_k$  les coordonnées du  $k^{\text{ième}}$  centroïde  $c_k$ , de la partition de mots. On note par  $(j)$  l'index du plus proche centroïde au mot  $w_j$ . Par exemple,  $\mu_{(j)}$  dénote le barycentre du plus proche centroïde associée au mot  $w_j$ .

### 3.2. Apprentissage de concepts

Nous définissons un concept comme un ensemble de mots qui est déterminé à partir de l'analyse des co-occurrences de mots dans les paragraphes. Pour découvrir ces ensembles nous utilisons les co-occurrences des mots dans les paragraphes. Chaque mot  $w$  de  $V$  est d'abord représenté par un vecteur  $\vec{w} = \langle n(w, i) \rangle_{i \in \{1, \dots, n\}}$  de dimension  $n$  caractérisant le nombre d'occurrences du mot  $w$  dans chaque paragraphe  $x_i$ . En se basant sur cette représentation mot-paragraphe, on utilise un algorithme de partitionnement sur les mots, l'algorithme des *X-moyennes* pour trouver les différentes classes de mots dans l'espace des paragraphes. Cet algorithme est une extension de l'algorithme classique des *K-moyennes* dans lequel le nombre de classes est estimé au lieu d'être fixé par l'utilisateur. Les mots appartenant aux mêmes ensembles (les classes de l'algorithme de partitionnement) auront la même représentation dans le nouvel espace. L'algorithme commence avec deux partitions et décide itérativement quand est-ce qu'il faut couper une classe en utilisant un critère d'information bayésien (CIB). Si en partitionnant, ce critère décroît, on garde alors l'ancienne partition. L'hypothèse sous jacente est que les mots qui co-occurrent fréquemment sont sémantiquement proches, c'est une hypothèse courante en recherche d'information, que l'on retrouve également dans d'autres méthodes de réduction de la dimension comme la méthode de projection "Latent Semantic Analysis" [DUM 98].

Par rapport à cette dernière qui est basée sur des combinaisons linéaires de mots difficilement interprétables, notre méthode permet de conserver pour chaque classe les mots représentatifs. Au-delà d'une simple réduction de dimension, cette approche permettra ensuite de regrouper des paragraphes similaires au niveau des concepts ainsi identifiés. Deux paragraphes peuvent être jugés similaires même s'ils ne possèdent que peu de mots en commun à partir du moment où ceux-ci caractérisent les mêmes concepts. Cette représentation sémantique, même si elle est relativement fruste, est considérablement plus riche que la représentation sac de mot des entités que l'on veut comparer. Elle permet également d'identifier très clairement quels sont les différents concepts présents dans un paragraphe.

Formellement, on fait l'hypothèse que les mots  $w$  sont générés indépendamment par un mélange de densités :

$$p(\vec{w}) = \sum_{k=1}^K \pi_k p(\vec{w} \mid c = k) \quad [1]$$

$K$  dénote le nombre de partitions trouvées et les  $\pi_k$  représentent les probabilités de classes. Les  $\vec{w}$  sont ici les mots représentés dans l'espace des paragraphes. Nous avons trouvé que la modélisation des densités  $p(\vec{w} \mid c = k)$  par des densités gaussiennes hypersphériques de matrice de covariance commune  $\Sigma = \sigma^2 I$  était un bon compromis entre l'efficacité et la complexité. Les estimateurs du maximum de vraisemblance des centroïdes  $\hat{\mu}_k$  et  $\hat{\sigma}^2$  sont :

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

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

En utilisant ce modèle de mélange, la log-vraisemblance du centroïde  $c_k$  est :

$$\hat{l}_m(c_k) = \sum_{w_j \in c_k} \left( \frac{1}{\sqrt{2\pi}\hat{\sigma}} - \frac{1}{2\hat{\sigma}^2} \|\vec{w}_j - \mu_k\|^2 + \log \pi_k \right) \quad [2]$$

Où,  $\|\cdot\|$  désigne la norme euclidienne. Le meilleur modèle ici est celui qui maximise le critère CIB défini par :

$$CIB = \sum_k \hat{l}_m(c_k) - \frac{p_m}{2} \log P \quad [3]$$

Où,  $p_m$  est le nombre de paramètres du modèle probabiliste i.e. les moyennes et variances des composantes du mélange.

X-moyennes débute avec un nombre maximum d'itérations  $T$  et deux partitions initiales obtenues avec l'algorithme K-moyennes avec  $K = 2$  sur tout le vocabulaire. Avec cette partition initiale, il utilise alors l'algorithme 2-moyennes sur chaque classe et vérifie si ce nouveau partitionnement fait croître CIB. Si c'est le cas, la classe initiale est remplacée par deux de ses fils. L'algorithme recherche sur le vocabulaire entier la meilleure classe à partitionner.

Dans les expériences détaillées dans la section 4, nous trouvons 217 classes de mots pour un vocabulaire de taille 16252. Le tableau 1 montre quelques une des classes trouvées, on peut vérifier sur cet exemple que chaque classe de mots peut être associée à un concept général. Dans la section suivante, on représente un paragraphe en utilisant les classes trouvées avec l'algorithme X-moyennes. Cette nouvelle caractérisation permet une représentation concise et efficace des paragraphes.

**Tableau 1.** Un exemple de classes de mots (concepts) trouvées avec l'agorithme X-moyennes

**Partition  $i$  :** video feature graphic format info print adobe acrobat reader  
connection browse html cooper valve animation modem printing

**Partition  $j$  :** world design history manufacturer usa staff list engine  
technique commercial personnel maintenance routine leader navigation crew  
repair part interior aircraft modification overhaul aviation flight

**Partition  $k$  :** belt cold check battery heater winter weather vehicle idea  
fluid tire hose driving filter brake antifreeze

**Partition  $l$  :** 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

### 3.3. Réduction de dimensionalité

Les segments de textes, ici les paragraphes, seront maintenant représentés dans l'espace des concepts. Pour cela, un paragraphe  $x_i$  sera caractérisé par un vecteur dans l'espace des concepts  $\vec{x}_i = \langle \bar{n}(c, i) \rangle_{c \in \{1, \dots, |C|\}}$  où la caractéristique  $\bar{n}(c, i)$  représente le nombre d'occurrences des mots du concept  $c$  dans le paragraphe  $x_i$  et  $|C|$  est le nombre de concepts  $c$  découverts. Les caractéristiques d'un paragraphe dans cette nouvelle représentation traduisent le degré de représentation de chaque concept dans ce paragraphe. Dans la dernière étape les paragraphes seront comparés dans ce nouvel espace de concepts.

En s'appuyant sur cette caractérisation, nous présentons, dans la section suivante, une technique de partitionnement basée sur l'approche de la vraisemblance classifiante pour segmenter un texte en classes de paragraphes.

### 3.4. Partitionnement de paragraphes

L'approche dite de vraisemblance classifiante (VC) est un formalisme général qui permet de formuler et justifier de façon unifiée un grand nombre d'algorithmes de partitionnement qui ont été proposés. La méthode spécifique de partitionnement utilisée n'est pas essentielle pour notre algorithme de segmentation, celle que nous proposons ci-dessous offre l'avantage de la simplicité. D'autres méthodes pourraient être utilisées. Dans notre cas, la vraisemblance classifiante définie sur les paragraphes est :

$$L_{VC} = \sum_{i=1}^n \sum_{k=1}^{\Omega} t_{ki} \log p(\vec{x}_i, y = k) \quad [4]$$

Où  $n$  est le nombre de paragraphes dans la collection,  $\Omega$  le nombre de classes (concepts) trouvées et  $t_{ki}$  un indicateur de classe qui vaut 1 si  $x_i$  appartient à la partition  $k$  et 0 sinon. Les paragraphes sont supposés être générés indépendamment suivant un mélange de densités :

$$p(\vec{x}) = \sum_{k=1}^{\Omega} p(y = k) p(\vec{x} | y = k) \quad [5]$$

Les paramètres du mélange sont estimés en maximisant le critère VC.

L'algorithme utilisé pour la vraisemblance classifiante est l'algorithme CEM [CEL 92]. C'est une technique itérative similaire à l'algorithme EM [DEM 77] sauf pour une étape additionnelle de classification où chaque exemple  $x_i$  est affecté à une et une seule composante du mélange. Cet algorithme est brièvement présenté ci-dessous. En suivant une approche classique pour l'estimation de densité dans les

---

#### Algorithm 1 CEM

---

*Initialisation* : Soit  $\Pi^{(0)}$  une partition initiale

Pour la  $j^{ime}$  itération,  $j \geq 0$  :

– E-step : Estimer les probabilités a posteriori de classes de l'appartenance de l'exemple  $x_i$  à  $\Pi_k$  ( $i = 1, \dots, n; k = 1, \dots, c$ ) :

$$E[t_{ki}^{(j)} | x_i; \Pi^{(j)}] = \frac{p(y^{(j)} = k) p(\vec{x}_i | y^{(j)} = k)}{\sum_{k=1}^{\Omega} p(y^{(j)} = k) p(\vec{x}_i | y^{(j)} = k)} \quad [6]$$

– C-step : Assigner chaque  $x_i$  à la partition  $\Pi_k^{(j+1)}$  qui maximise cette probabilité a posteriori.

– M-step : Estimer les nouveaux paramètres du modèle de mélange qui maximise  $L_{VC}$ .

---

textes, nous supposons que pour chaque composante du mélange  $p(\vec{x}_i | y^{(j)} = k)$ , les caractéristiques de  $v$  sont indépendantes et la densité sera estimée par un modèle Naïve Bayes. Les paramètres de ce modèle sont l'ensemble des probabilités de mélanges  $\pi_k = p(y = k)$  et des coefficients des lois binomiales de chacune des caractéristiques  $p_{ck}$  qui denotent la probabilité d'apparition d'une occurrence du

**Tableau 2.** Les caractéristiques de la collection *7sectors*

Classe	taille	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

concept  $c$  dans un paragraphe. Sous ces hypothèses,  $p(\vec{x} | y = k) \equiv \prod_{c=1}^{|C|} p_{ck}^{\bar{n}(c,x)}$  avec  $\bar{n}(c, x)$  qui est le nombre d'occurrences du concept  $c$  dans le paragraphe  $\vec{x}$ . En dérivant (4) par rapport aux  $\pi_k$  et  $p_{ck}$  et en introduisant des multiplicateurs de Lagrange pour prendre en compte les contraintes suivantes sur les paramètres  $\sum_k \pi_k = 1$  et  $\forall k, \sum_c p_{ck} = 1$ , les estimateurs du maximum de vraisemblance pour  $\pi_k$  et  $p_{ck}$  sont :

$$\pi_k = \frac{\sum_{i=1}^n t_{ki} + 1}{n + \Omega} \quad [7]$$

$$p_{ck} = \frac{\sum_{i=1}^n t_{ki} \bar{n}(c, x_i) + 1}{|x_i| + |C|} \quad [8]$$

Où,  $|x_i| = \sum_{c=1}^{|C|} \bar{n}(c, x_i), \forall i$ . Cette étape réalise donc une classification automatique des paragraphe dans l'espace des concepts préalablement appris.

## 4. Résultats Expérimentaux

### 4.1. Le corpus

En suivant la proposition de [STO 02] concernant l'évaluation des algorithmes de segmentation thématique, nous avons construit une collection de test en concaténant des documents d'une collection de données. Nous avons choisi dans nos expériences d'utiliser la collection de données<sup>1</sup> *7sectors*. Cette collection est issue du projet Web->KB<sup>2</sup> et elle est constituée de 3417 documents *html*. Nous avons pris chaque document comme un paragraphe dans nos expériences, il y a 17 phrases en moyenne dans chaque document de cette collection et prendre un document pour un paragraphe n'est pas irréaliste. Après avoir filtré le texte en enlevant les balises *html*, notre prétraitement convertit les majuscules en minuscules et ignore les caractères non alpha-numériques. Nous enlevons également les mots d'un anti-dictionnaire ainsi que les mots qui apparaissent dans moins de 3 paragraphes. Le vocabulaire obtenu se résume après ces filtrages à 16252 mots. Le tableau 2 récapitule les caractéristiques de cette collection.

1. <http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/bootstrappingIE/7sectors.tar.gz>

2. <http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-11/www/wwkb/>

Nous avons testé trois algorithmes - notre méthode de segmentation utilisant la représentation des paragraphes dans l'espace des concepts (RPC avec CEM), l'algorithme CEM avec la représentation simple de sac-de-mots (RPS avec CEM) et l'algorithme de référence de Salton. Pour comparer les résultats de ces différentes approches non-supervisées, nous fixons le nombre de partitions de paragraphes à 7, ce qui correspond exactement au nombre de classes thématiques dans la collection. L'expérience a pour but de comparer les méthodes à nombre de classes de paragraphes (les thèmes) fixe. En situation réelle, on ne connaît pas a priori les thèmes présents dans le corpus et il faut déterminer le nombre de classes automatiquement. La méthode d'évaluation des résultats devient alors plus complexe et également plus subjective. Il est difficile par exemple de comparer objectivement des partitions à nombre de classes différents (cela ne pose pas de problème formellement, mais le résultat est difficilement interprétable). C'est pourquoi nous avons adopté ici cette approche directe pour l'évaluation.

#### 4.2. Mesure d'évaluation

Les mesures d'évaluations utilisées sont les micro-moyennes de précision et de rappel. Pour l'estimation de ces mesures [SLO 02], nous avons d'abord affecté tous les paragraphes d'une classe donnée au concept majoritaire dans cette classe. Pour chaque concept  $c$ , nous avons alors estimé les quantités suivantes :

- $\alpha(c)$  : Le nombre de paragraphes correctement affectés à  $c$ ,
- $\beta(c)$  : Le nombre de paragraphes incorrectement affectés à  $c$ ,
- $\gamma(c)$  : Le nombre de paragraphes incorrectement non affectés à  $c$ .

Les micro-moyennes pour la précision et le rappel sont définies comme suit :

$$\text{Précision} = \frac{\sum_c \alpha(c)}{\sum_c \alpha(c) + \beta(c)} \quad \text{Rappel} = \frac{\sum_c \alpha(c)}{\sum_c \alpha(c) + \gamma(c)}$$

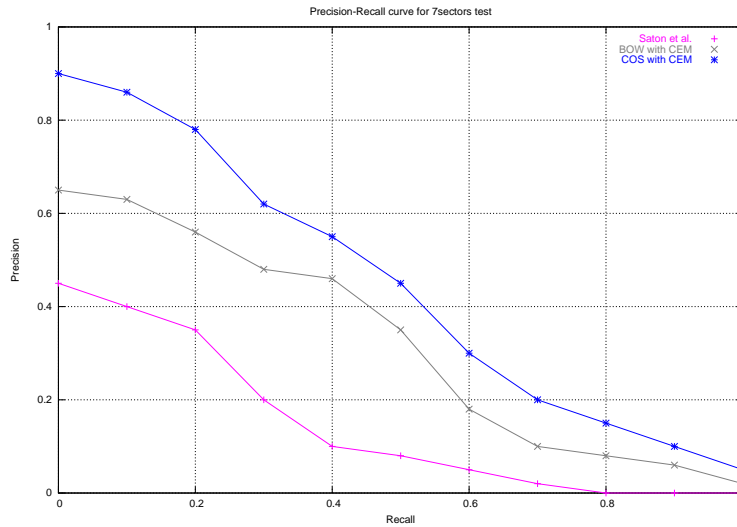
#### 4.3. Résultats

Dans le tableau 3, nous présentons les résultats obtenus sur le corpus `7sectors`. Notre système de segmentation (RPC avec CEM) obtient des précisions supérieures à celles des deux autres systèmes. Bien que le nombre de classes thématiques soit fixé a priori, le système de Salton a trouvé une classe thématique très majoritaire en y rassemblant à peu près tous les paragraphes. En comparant, les modèles de segmentation basés sur le CEM, il apparaît que le système RPC permet non seulement une représentation plus concise des paragraphes mais caractérisent aussi les paragraphes d'une manière plus efficace que la représentation sac de mots au sens de la mesure utilisée. Les valeurs portées dans ce tableau correspondent au point "précision = rappel". Certains thèmes ne sont pas du tout identifiés dans les tests réalisés (e.g. energy, health, utilities). Il s'agit des thèmes les plus faiblement représentés dans le corpus. Cela ne traduit pas une limitation des méthodes employées mais un biais de l'expérience réalisée. L'étiquetage des classes de concepts favorise celles qui sont les plus représentées et fait disparaître les autres. Les différentes classes du corpus ont des intersections non nulles et des paragraphes de différentes classes, qui sont des entités courtes en comparaison avec les documents qui définissent les classes thématiques du corpus, se retrouvent classés ensemble. Pour obtenir des classes plus homogènes, il faut employer un nombre de classes plus important que l'on peut ensuite examiner manuellement pour leur affecter des étiquettes. Cela rejoint le commentaire de la section 4.1 sur la difficulté de l'évaluation dans ces conditions. On peut remarquer que la méthode proposée permet d'identifier une classe supplémentaire (transport.) par rapport à la même méthode employée sur une représentation sac de mots. Il faut bien noter que ces résultats correspondent à des performances moyennes et que l'algorithme peut être employé à d'autres points de fonctionnement que celui où "précision = rappel".

**Tableau 3.** Micro-moyenne de Précision pour les différents systèmes de segmentation sur la base 7sectors.

	RPC avec CEM			RPS avec 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
<b>Précision moyenne</b>	<b>49.05</b>			<b>42.25</b>			<b>24.17</b>		

Sur la figure 1, nous présentons les courbes de rappel et de précision pour les trois systèmes. Ces courbes confirment les résultats précédents en ce qui concerne les performances relatives des trois systèmes. En fonction de l'application pour laquelle on veut utiliser la segmentation, on se placera à différents points de cette courbe. Pour du résumé par exemple, on aura tendance à favoriser les précisions élevées, on voit sur la courbe qu'à un rappel de 0,1 ou 0,2 on obtient des précisions très élevées avec la méthode proposée.

**Figure 1.** Les courbes de Précision-Rappel pour les différentes méthodes de segmentation sur la base 7sectors.

## 5. Conclusion

Nous avons proposé une nouvelle approche non-supervisée pour entraîner un système pour la segmentation thématique de textes basé sur l'extraction de paragraphes. Dans cette approche, les paragraphes sont codés dans un espace de concepts permettant une réduction importante de la dimension par rapport à l'approche sac-de-mots. Notre méthode a été comparée au système de segmentation de Salton et à un autre système non-supervisé, basé sur l'algorithme CEM, mais utilisant la représentation sac-de-mots. Le système proposé obtient de meilleurs résultats dans le cadre de notre évaluation. Les extensions de ce travail concernent l'évaluation sur des corpus plus larges et hétérogènes, ainsi que la mise en oeuvre d'évaluations plus complexes où le nombre de thèmes n'est pas fixé à l'avance.

## Remerciements

Ce travail a été sponsorisé en parti par le programme IST de la communauté Européenne, sous le réseau d'excellence de PASCAL, IST-2002-506778. Cette publication reflète seulement les points de vue des auteurs.

## 6. Bibliographie

- [BEE 97] BEEFERMAN D., BERGER A., J. L., « Text segmentation using exponential models », *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, 1997, p. 35-46.
- [BEL 01] BELLOT P., EL-BÈZE M., « Classification et segmentation de textes par arbres de décision », *Technique et Science Informatiques (TSI)*, vol. 20, n° 3, 2001, p. 397-424, Editions Hermès.
- [BIG 98] BIGI B., DE MORI R., EL-BÈZE M., SPRIET T., « Detecting topic shifts using a cache memory », *Proceedings of the fifth Conference on Spoken Language Processing*, 1998.
- [CEL 92] CELEUX G., GOVAËRT G., « A Classification EM algorithm for clustering ans two stochastic versions », *Computational Statistics and Data Analysis*, vol. 14, n° 3, 1992, p. 315-332.
- [DEM 77] DEMPSTER A., LAIRD N., RUBIN D., « Maximum Likelihood from incomplete data using the EM algorithm », *Journal of the Royal Statististical Society*, vol. 39(B), 1977.
- [DUM 98] DUMAIS S., FURNAS G., LANDAUER T., DEERWESTER S., HARSHMAN R., « Using Latent Semantic Analysis to Improve Access to Textual Information », *Proceedings of the Conference on Human Factors in Computing Systems CHI'98*, 1998.
- [HEA 97] HEARST M., « TextTiling : Segmenting text into multi-paragraph subtopic passages », *Computational Linguistics*, , 1997, p. 33-64.
- [KAN 98] KAN M. Y., KLAVANS J., « Linear segmentation and segment significance », *Proceedings of the 6<sup>th</sup> workshop on very large corpora (WVLC-98)*, 1998, p. 197-205.
- [KOZ 93] KOZIMA H., « Text segmentation based on similarity between words », *Proceedings on ACL*, 1993, p. 286-288.
- [LIT 95] LITMAN D., PASSONEAU R., « Combining multiple knowledge sources for discourse segmentation », *Proceedings on ACL*, 1995, p. 108-115.
- [MIL 90] MILLER G. A., BECKWITH R., FELLBAUM C., GROSS D., MILLER K., « Five Papers on WordNet », Technical report, 1990, Cognitive Science Laboratory, Princeton University.
- [PON 97] PONTE J., CROFT W., « Text segmentation by topic », *Proceedings of the First European conference on Research and Advanced Technology for Digital Libraries*, 1997, p. 120-129.
- [SAL 96] SALTON G., SINGHAL A., BUCKLEY C., MITRA M., « Automatic Text decomposition using text segments and text themes », *Proceedings of the 7<sup>th</sup> ACM conference on hypertext*, 1996.
- [SLO 02] SLONIM N., FRIEDMAN N., TISHBY N., « Unsupervised Document Classification using Sequential Information Maximization », *Proceedings of the 25<sup>th</sup> ACM SIGIR Conference*, 2002, p. 129-136.

- [STO 02] STOKES N., CARTHY J., SMEATON A., « Segmenting Broadcast News streams using lexical chains », *Proceedings of 1<sup>th</sup> AI researchers symposium*, 2002, p. 145-154.
- [WAL 91] WALKER W., « Redundancy in collaborative dialogue », *Actes of AAAI symposium on Discourse Structure in Natural Language Understanding and Generation*, 1991.
- [YAM 98] YAMRON J., CARP I., GILLICK L., LOWE P., VAN MULBREGT P., « A hidden markov model approach to text segmentation and event tracking », *IEEE ICASSP*, 1998.