Unsupervised Text Segmentation using LDA and MCMC

Kaimin Yu **Zhe Li** **Genliang Guan** **Zhiyong Wang** **David Feng**

School of Information Technologies
University of Sydney
NSW, 2006, Australia
Email: yu.kaimin,zhli8662,genliang.guan,zhiyong.wang,dagan.feng@sydney.edu.au

Abstract

In this paper, we propose a data driven approach to text segmentation, while most of the existing unsupervised methods determine segmentation boundaries by empirically exploring similarity measurement between adjacent units (e.g. sentences). Firstly, we train a latent Dirichlet allocation (LDA) model with the large scale Wikipedia Corpus to avoid the problem of vocabulary mismatch, which makes our approach domain-independent. Secondly, each segment unit is represented with a distribution of the topics, instead of a set of word tokens. Finally, a text input is modeled as a sequence of segment units and Markov Chain Monte Carlo technique is employed to decide the appropriate boundaries. The major advantage of using MCMC is its ability to detect both strong and weak boundaries. Experimental results demonstrate that our proposed approach achieve promising results on a widely used benchmark dataset when compared with the state-of-the-art methods.

Keywords: Text Segmentation, Topic Model, LDA, Markov Chain Monte Carlo(MCMC), Data Driven

1 Introduction

Text segmentation is to divide a given text data into semantically relevant and coherent segments. It is normally consider as an important prerequisite step for other high level semantic text analysis tasks, such as summarization and information retrieval. For example in the context of information retrieval, web pages often vary in length and content, while some short web page may focus on one topic, web pages that contain lengthy documents are likely to address multiple topics. By dividing a document into topic coherent segments, search engines can index the resulting segments based on the topics which will allow users to quickly access information of interest within a lengthy document.

Various unsupervised and supervised approaches have been proposed for text segmentation. In comparison with supervised segmentation algorithms, unsupervised methods require less domain specific knowledge (e.g. *welcome* and *next* in the transcriptions of TV news programs) and more suitable for domain-independent applications. Most of the existing methods in this category utilize lexical cohesion among segment units (e.g. sentences) (Choi et al. 2001).These approaches often rely on some heuristic rules (e.g. repetition) to derive lexical cohesion. Recently Misra *et al.* proposed to employ topic modelling techniques for text segmentation (Misra et al. 2009). The well established latent Dirichlet allocation (LDA) model was utilized to learn hidden topics in a generative and unsupervised manner and each document is represented as a distribution of topics. Therefore, lexical cohesion is replaced with similarity measurement in terms of topic distribution in calculating pair-wise path scores. In addition, the segments obtained are associated (or labelled) with topic information.

Rather than calculate cumulative scores of potential paths with topic distributions, we formulate text segmentation with a probabilistic problem which can be solved with the unsupervised Markov Chain Monte Carlo (MCMC) technique. As indicated in (Zhai & Shah 2006), MCMC is able to detect both the strong and weak boundaries (Zhai & Shah 2006).

Due to the small training dataset used by (Misra et al. 2009), they had to deal with the problem of vocabulary mismatch (i.e. the difference between vocabularies of training dataset and test dataset). In this work, we investigate the impact of using a large scale web corpus, Wikepedia Corpus [1]. It is expected that more representative topics can be discovered from such a large scale corpus and eventually the problem of vocabulary mismatch can be eliminated. Our experimental results indicate that larger dataset help achieve better segmentation performance.

The rest of paper is organized as follows. The related work is reviewed in Section 2. Sections 3 and 4 describe the proposed unsupervised text segmentation method using LDA and MCMC. In Section 5, we compare our method with several state-of-art text segmentation methods and present the experimental results. Finally, conclusions are given in Section 6.

2 Related Work

Linear text segmentation has attracted a significant attention in the field due to its importance in natural language processing tasks, such as information extraction and text summarization. Early approaches (Passonneau & Litman 1997, Beeferman et al. 1999) often exploit the linguistic information such as cue phrases, syntax or lexical features. They assume certain words or phrases can be used to detect the segment boundaries. For example, in TV new programs, cue phrases like "hello and welcome to" and "good evening I'm" typically appears in the beginning of news stories. Conversely, cue phrases, including "stay

[1] http://en.wikipedia.org/wiki/Wikipedia:Database_download

with us", "when we come back" and "weather forecast is next", often indicates the end of a segment. While cue phrases may convey the document structures, they are normally specific for a type of data and cannot be generalized to other application domains. For each new application, a new set of cue phrases are required to be identified which can be very time consuming and cost prohibitive (Misra et al. 2009, 2010).

The most dominant direction in text segmentation is based on the lexical cohesions (Hearst 1997, Choi 2000, Utiyama & Isahara 2001, Fragkou et al. 2004, Malioutov & Barzilay 2006). These approaches are built around the fact that related or similar words tend to be repeated in topically coherent segments and a change in the vocabulary often indicates segment boundaries. Such approaches normally do not require supervised training, hence they can be applied to any text form any domain. TextTiling (Hearst 1997) is one of the most influential approach in this category. It works by first dividing a document into blocks of fixed number of words which is usually 3-5 sentence long, and the similarity of adjacent blocks is measured based on cosine similarity. The resulting sequence of similarity values is then graphed and smoothed. The local maxima of the word similarity curve indicates that the adajcent blocks cohere well, whereas the local minima is the point of low lexical cohesion and being regards as a potential segment boundary candidate. However, the numerical value of the similarity is prone to local extrema which has shown to be unreliable (Choi et al. 2001). Choi (2000) replaced the numerical similarity values with its rank in the local region, and used divisive clustering for segmentation in their C99 algorithm. Other techniques have also been used for segmentation. Fragkou et al. (2004) proposed a dynamic programming algorithm to perform the text segmentation by global minimizing the segmentation cost. In order to address the poor segmentation performance caused by smooth topic transitions (weak boundaries), Malioutov & Barzilay (2006) represent a text document as a weighted undirected graph and formalized the text segmentation task as graph partition sloved using normalized cut. Kazantseva & S. (2011) proposed to utilize Affinity Propagation clustering algorithm to locate the segment boundaries and segment centers.

Recently topic models are used to compute the similarity. By adopting topic models, the similarity are measured not only based on the exact word repetitions, but also the relations of related words. Choi et al. (2001) applied Latent Semantic Analysis (LSA) (Landauer et al. 1998) in the C99 (Choi 2000) algorithm to measure the sentence similarities where a sentence is represented by the sum of the LSA feature vectors. Their experimental results show that the LSA based similarity measures can significantly outperform the cosine metric used in the original C99 algorithm (Choi et al. 2001). Latent Dirichlet Allocation (LDA) topic models are also exploited by a group of other researchers(Sun et al. 2008, Misra et al. 2009, Riedl & Biemann 2012). Misra et al. (2009, 2011) used the topics discovered by LDA to compute the log-likelihood of each possible segment. The log-likelihood was then used as a score in the dynamic programming algorithm to recover the segmentation from the path that yields the highest log-likelihood (Misra et al. 2009). Sun et al. (2008) used kernel function to measure how much two segments share the same latent topic and dynamic programming for segments selection. Riedl & Biemann (2012) proposed the TopicTitling algorithm which uses topics obtained by LDA model in a similar fashion as TextTiling uses words.

In our work, LDA is used to compute the pairwise sentence similarity as it is shown to be very effective (Sun et al. 2008, Misra et al. 2009, Riedl & Biemann 2012). Unlike previous approaches, we use the data-driven Markov Chain Monte Carlo (MCMC) techniques to discover the segment boundaries. The major advantage is that MCMC is able to detect both the strong and weak boundaries.

3 Topic modeling with LDA

LDA is a probabilistic generative model to explore the topics of a set of documents. It assumes that each document can be represented by a distribution of the topics and each topic has its underlying multinomial distribution over the vocabulary (Blei et al. 2003). Note that LDA ignores the word orders which means that the words in a document are interchangeable. For example, "topic modeling with LDA" and "LDA with topic modeling" are viewed as completely equivalent by the LDA model.

Given a set of topics t_i, $i = 1, \ldots, N_T$ and a vocabulary $W = \{w_i | i = 1, \ldots, N_W\}$, LDA assumes a document d can be produced as follows. First, a distribution β_t over the vocabulary is drawn from a Dirichlet distribution for each topic t. Second, a topic distribution θ_d for d is randomly drawn from a Dirichlet distribution. Finally, each word w_i in the document d is generated by selecting a topic according to the topic distribution θ_d and then randomly choosing a word from the chose topic based on the word distribution for the topic β_t. Formally, the probability of the i^{th} word is as follows (Misra et al. 2009):

$$\begin{aligned} P(\omega_i \mid \theta_d, \beta) &= \sum_{t=1}^{N_T} P(t_i = t \mid \theta_d) P(\omega_i \mid t_i, \beta) \\ &= \sum_{t=1}^{N_T} \theta_{dt} \beta_{tw} \end{aligned} \tag{1}$$

where θ_{dt} is the probability of using the topic t in the document d and β_{tw} is the probability of using the word w in the topic t.

The topic distribution θ for each document d and the word distribution β for each topic t are the parameters that need to be inferred from a corpus. Gibbs sampling is used to estimated these two model parameters as follows (Griffiths & Steyvers 2004):

$$\theta_{dt} = \frac{K_{dt} + \alpha}{\sum_{k=1}^{N_T} K_{dk} + N_T \alpha} \tag{2}$$

$$\beta_{tw} = \frac{J_{tw} + \lambda}{\sum_{k=1}^{N_W} J_{tk} + N_W \lambda} \tag{3}$$

where K_{dt} is the total number of words in the document d that are assigned to topic t, J_{tw} is the number of times a word w is assigned to a topic t, α and λ are Dirichlet priors.

After obtaining the word distribution β_{tw} for each latent topic t, the topic distribution of an unknown document can be estimated iteratively as (Misra et al. 2008):

$$\theta_{dt}^{n+1} = \frac{1}{L_d} \sum_{w=1}^{N_W} \frac{C_{d\omega} \theta_{dt}^{(n)} \beta_{tw}}{\sum_{t'=1}^{N_T} \theta_{dt'}^{(n)} \beta_{t'w}} \tag{4}$$

where $\theta_{dt}^{(n)}$ is the value of θ_{dt} at the nth iteration, l_d is the number of words in the document d that are presented in the training vocabulary W, and $C_{d\omega}$ is the count of word ω in d. It should be noted that the words in d but not in W are ignored in this process. Given the topic distribution θ_d for a document d and the word distribution β for all the discovered latent topics, the likelihood of document d can be calculated as:

$$P(C_d \mid \theta_d, \beta) = \prod_{\omega=1}^{N_W} \left(\sum_{t=1}^{N_T} \theta_{dt}\beta_{tw} \right). \quad (5)$$

In this paper, the LDA model is trained using a large scale web corpus, Wikepeida Corpus. It is expected that discovered topics can be more general with boarder applications. We then apply the learned LDA model to a test document for measuring the pairwise sentence similarities. Specifically, we compute the topic distribution for each sentence and measures their Euclidean distance as follows:

$$D(s_i, s_{i+1}) = \Bigg[\Big(p(t_1 \mid s_i) - p(t_1 \mid s_{i+1}) \Big)^2 + \ldots + \Big(p(t_n \mid s_i) - p(t_n \mid s_{i+1}) \Big)^2 \Bigg]^{\frac{1}{2}} \quad (6)$$

where $\{t_i \mid i = 1 \ldots n\}$ is the latent topics obtained by the trained LDA model. Once the pairwise sentence similarity matrix is built, the linear text segmentation problem can be solved by the Markov Chain Monte Carlo technique as discussed in Section 4.

4 Boundary detection with MCMC

Linear text segmentation is a process of partitioning a given document into meaningful segments, such that each segment is coherent about a specific topic and consecutive segments are about different topics. In this paper, we consider sentence as the smallest unit that forms a document, hence a segment consists of one or more sentences. Let k denotes the potential number of segments in a document and θ_k denotes their corresponding boundary locations, the general Metropolis-Hasting-Green algorithm (Green 1995) is employed to estimate these two parameter as follows, where $x = k, \theta_k$ and $\pi(x)$ denotes the posterior probabilities of x:

1) The parameter x_0 is initialized.

2) The followings are conducted in each iteration i.

3) Generate Th_α from $Uni[0, 1]$.

4) Create a new parameter x'_{i-1} based on x_{i-1} with a diffusion or jump.

5) Calculate the radio $\alpha(x_{i-1}, x'_{i-1})$ as:

$$\alpha(x_{i-1}, x'_{i-1}) = min\{1, \frac{\pi(x'_{i-1})q(x'_{i-1}, x_{i-1})}{\pi(x_{i-1})q(x_{i-1}, x'_{i-1})}\} \quad (7)$$

6) Update $x_i = x'_{i-1}$, if $\alpha > Th_\alpha$. Otherwise, set $x_i = x_{i-1}$

In Equation 7, $q(x, x')$ is the transition probability from state x to x'. Such probability between two states is dependent on the updates types and should be reversible. As in (Zhai & Shah 2006), there are two types of updates that are diffusion and jump. The diffusion update simulates the shifting of boundaries between two adjacent text segments, hence the dimension of the parameter θ_k does not change. The jump update simulates a pair of reversed actions: split and jump. Split divides a text segment into two parts which increase the dimension of θ_k by 1, while merge combines two adjacent text segments into one thus reducing the dimension of θ_k by 1. The details will be discussed in the following.

4.1 Diffusion

Diffusion is the process of updating the location of the boundary between two adjacent text segments. It uniformly randomly selects a segment boundary and draws a new boundary from a 1D normal distribution with the mean at its original position. Assume t' denotes the new location of the boundary and t denotes its original position, the probability of drawing t' from t can then be calculated as (Zhai & Shah 2006):

$$p(t') = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(t'-t)^2}{2\sigma^2} \right) \Big(I(t') \Big) \quad (8)$$

where σ is the standard deviation of the movement, and $I(t')$ is a indicator function which is 1 only if the new boundary is within the correct range of the updated segment. Then the forward transition probability for the shift update becomes $q(x, x') = 1/(k-1))p(t')$, and the backward transition probability is $q(x', x) = (1/(k-1))(1 - p(t'))$, where k is the number of segments.

4.2 Jump

The jump update consists of two reversed actions: split and merge. Split divides a original segment $S_m = \{s_m^1, \ldots, s_m^n\}$ into two new segments $S'_m = \{s_m^1, \ldots, s_m^{t-1}\}$ and $S'_{m+1} = \{s_m^t, \ldots, s_m^n\}$, where s_m^t is the new boundary. The data-driven technique (Zhai & Shah 2006) is used to propose the new boundary. We assume uniform probability for selecting scene S_m, the new boundary location t is selected to maximize the likelihood of the new segments as follows:

$$t = \arg\max(\mathbb{L}(S'_m \mid f'_m) + \mathbb{L}(S'_{m+1} \mid f'_{m+1})) \quad (9)$$

where $\mathbb{L}(S'_m \mid f'_m)$ and $\mathbb{L}(S'_{m+1} \mid f'_{m+1})$ are the likelihood of the two new segments S'_m and S'_{m+1}, f is the features used to measure the sentence similarity. The transition probability for split can then be calculated as:

$$q(x, x') = \frac{1}{k}\mathbb{L}(S'_m \mid f'_m)\mathbb{L}(S'_{m+1} \mid f'_{m+1}) \quad (10)$$

Merge is the reversed update of split which combines two adjacent segments into one. As in (Zhai & Shah 2006), we assume uniform probability for selecting segment S_m and combine it with S_{m+1} to form a new segment S'_m. The transition probability can then be easily obtained as follows:

$$q(x, x') = \frac{1}{k-1}\mathbb{L}(S'_m \mid f'_m). \quad (11)$$

4.3 Posterior Probability

The posterior probability of the two parameters k and θ_k is:

$$p(k, \theta_k \mid y) \propto \mathbb{L}(y \mid k)p(\theta_k \mid k)p(k) \quad (12)$$

```
1  ==========
2  Payne dismounted in Madison Place and handed the reins to Herold .
3  There was a fog , which increased the darkness of the night .
4  Two gas lamps were no more than a misleading glow .
5  He might have been anywhere or nowhere .
6  The pretence was that he was delivering a prescription from Dr. Verdi .
7  ==========
8  Note : Directions are written for those who have had previous experience .
9  Instructions for preparing clay , drying , glazing and firing are not given .
10 Equipment : Basic pottery studio equipment .
11 Wooden butter molds and cookie presses .
12 ==========
```

Figure 1: Two sample segments from the Choi's "3-5" dataset

where y is the feature selected for computing the sentence similarities, $\mathbb{L}(y \mid k)$ is the overall data likelihood given θ_k, $p(\theta_k \mid k)$ is the conditional probability for the boundary locations θ_k given k, and $p(k)$ is the prior probability for the number of segments.

As discussed before, different text segments are about different topics. Hence we can assume that each segment is independent from other, and the over data likelihood can be calculated as (Zhai & Shah 2006):

$$\mathbb{L}(y \mid \theta_k) = \left(\prod_{m=1}^{L} \mathbb{L}(y_m \mid f_m) \right)^{\frac{1}{L}}. \tag{13}$$

$\mathbb{L}(y_m \mid f_m)$ is the individual likelihood of data y_m in segment S_m and it is computed as the average of the pairwise similarity value of the sentences within S_m:

$$\mathbb{L}(y_m \mid f_m) = avg(\mathbb{M}(a : b, a : b)) \tag{14}$$

where $\mathbb{M}$ is the pairwise sentence similarity matrix obtained using the LDA model, a and b are the first and last sentence in S_m respectively.

The conditional probability for the boundary locations θ_k given k is defined in terms of the combinations as (Zhai & Shah 2006):

$$p(\theta_k \mid k) = \frac{(k-1)!(T-k)!}{(T-1)!} \tag{15}$$

where T is the total number of sentences in the given document.

As in (Zhai & Shah 2006), we assume the number of segments is drawn from a Poisson distribution as it models the number of incidents happening in a unit time interval. Hence, the model prior is calculated as:

$$p(k) = e^{-\lambda} \frac{\lambda^k}{k!} I(k) \tag{16}$$

where I(k) is an indicator function which equals to 1 if $1 \leq k \leq k_{max}$ and k_{max} is parameter that can be tuned based on the categories and length of the documents of interest.

5 Experiments

5.1 Experimental Settings

Our experiments were carried out with the widely used Choi's dataset (Choi et al. 2001). The Choi's dataset used in our experiments consists of 300 documents. Each document consists of ten text segments, where each segment is comprised of the first "n" sentences selected from an article in the Brown corpus. The successive segments within a document are corresponding to different topics. The Choi's dataset is divided into three subsets (namely "3-5", "6-8" and "9-11") based on the lengths of text segments "n". For example, the Choi's "3-5" dataset contains the segment with the length of 3 to 5 sentences. Figure 1 shows two successive segments from the "3-5" dataset. In order to investigate the impact of the training data size on segmentation performance, we also created 3 different datasets (namely A, B, and C) for training the LDA model by sampling the Wikipedia Corpus every 100, 50, and 10 entries, respectively.

Following previous research (Griffiths & Steyvers 2004), we set the Dirichlet priors (α and β) of the LDA model to (1 and 0.01), the number of topic to 200 (after a number of trials from 10 to 500), and the number of iterations to 600 (after a number of trials from 100 to 2000). For MCMC technique, the shifting distance variance is set to 3, the number of independent Markov chain to 200, and the iteration for each chain to 1000.

The evaluation protocol is the standard P_k (probabilistic error metric) which is the probability that two randomly drawn sentences which are K sentences apart are classified incorrectly. The higher value of P_k indicates lower accuracy in text segmentation. Compared to the conventional precision and recall measures, P_k penalizes near misses less than pure false positive and false negative, hence more accurately reflecting the segmentation performance.

5.2 Results and Discussions

5.2.1 Impact the of the size of the training dataset

The impact of the length of the text segments and the size of the training corpus on the segmentation performance is studied. As show in Figure 2, the segmentation performance consistently increases when the segment size increases from "3-5" to "9-11", which suggests that longer segments allow a more reliable estimation of the topic distribution by the LDA model. Moreover, it is observed that the larger the training corpus, the better segmentation performance can be obtained. As discussed in Section 3, during the estimation process, LDA drops the words which do not appear in the training process. Hence if there is a significant vocabulary mismatch between the training and testing data, potentially a large amount of words in the testing document will be dropped which can result in a great reduction of information thus affecting the segmentation performance. To demonstrate this, we trained our $LDA+MCMC$ model us-

ing three Wikipedia corpus of different sizes, namely A(smallest), B(middle), and C(largest). The result is shown in the last three rows in Table. 1. As can be seen, the model $LDA+MCMC(C)$ trained using the largest corpus C obtains the best performance due to the fact that the vocabulary size increases proportional to the size of the training corpus, thus it has a better chance to cover the testing vocabularies.

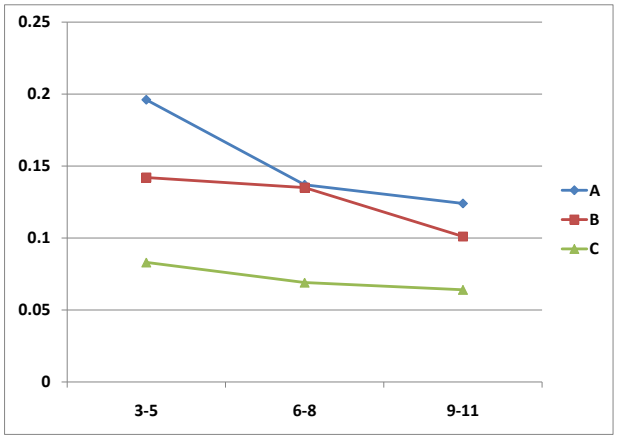

Figure 2: The impact of the length of the segment and the size of the training dataset on segmentation performance. The experiment is performed on the three Choi's subset, namely "3-5", "6-8" and "9-11". The lower the result, the better the performance.

5.2.2 Comparison with the state of the art

Our approach is also compared with the other state of the art methods. As shown in Table 1 where methods are sorted in chronicle order, our proposed approach achieves promising results benchmarked with the Choi's dataset. Specifically, our approach performs better than Unadapted LDA approach (Misra et al. 2009), which indicates the contribution from the MCMC technique. Though LDA (Adapted) approach achieves better result than our method, part of the Choi's dataset is required for training the LDA model to avoid the problem of vocabulary mismatch. Compared with the JSeg approach (Nguyen et al. 2011) which utilizes non-systematic relation in lexical cohesion, our approach also demonstrates better segmentation accuracy. JSegT approach further improves the segmentation performance when topic based similarity is combined with lexical distance (with empirically set combination weight). Interestingly, we are not able to achieve the similar gain when taking such combination into our MCMC based approach. It is worthwhile to investigate the fusion of different similarity measurements in the MCMC framework.

Methods	3-5	6-8	9-11	Avg
JTextTile	0.473	0.513	0.533	0.506
C99	0.115	0.104	0.112	0.110
TextSeg	0.090	0.070	0.050	0.070
MinCutSeg	0.340	0.241	0.174	0.252
LDA (Unadapted)	0.230	0.158	0.144	0.177
LDA (Adapted)	0.022	0.023	0.041	0.029
JSeg	0.091	0.107	0.121	0.106
JSegT	0.020	0.030	0.046	0.032
LDA+MCMC	**0.083**	**0.069**	**0.064**	**0.072**

Table 1: Comparison of segmentation performance on the Choi's dataset

6 Conclusions

We present an approach to text segmentation by combining the LDA model and the MCMC technique. Both methods are unsupervised and data driven, which makes our approach domain-independent. Our approach also achieve promising results on the benchmark dataset, when compared with the state-of-the-art methods. In the future, we will investigate the close integration of LDA and MCMC and further evaluate the proposed approach with topic models obtained from different datasets.

References

Beeferman, D., Berger, A. & Lafferty, J. (1999), 'Statistical models for text segmentation', *Machine learning* **34**(1), 177–210.

Blei, D., Ng, A. & Jordan, M. (2003), 'Latent Dirichlet Allocation', *Journal of Machine Learning Research* **3**, 993–1022.

Choi, F. (2000), Advances in domain independent linear text segmentation, *in* 'Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference', Morgan Kaufmann Publishers Inc., pp. 26–33.

Choi, F., Wiemer-Hastings, P. & Moore, J. (2001), Latent semantic analysis for text segmentation, *in* 'Proceedings of EMNLP', pp. 109–117.

Fragkou, P., Petridis, V. & Kehagias, A. (2004), 'A dynamic programming algorithm for linear text segmentation', *Journal of Intelligent Information Systems* **23**(2), 179–197.

Green, P. (1995), 'Reversible jump markov chain monte carlo computation and bayesian model determination', *Biometrika* **82**(4), 711.

Griffiths, T. & Steyvers, M. (2004), 'Finding scientific topics', *Proceedings of the National Academy of Sciences of the United States of America* **101**(Suppl 1), 5228.

Hearst, M. (1997), 'TextTilling: Segmenting texts into multi-paragraph subtopic passages', *Computational Linguistics* **23**(1), 33–64.

Kazantseva, A. & S., S. (2011), Linear Text Segmentation Using Affinity Propagation, *in* 'Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing'.

Landauer, T., Foltz, P. & Laham, D. (1998), 'An Introduction to Latent Semantic Analysis', *Discourse Processes* **25**(2-3), 259–284.

Malioutov, I. & Barzilay, R. (2006), Minimum cut model for spoken lecture segmentation, *in* 'Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics', Association for Computational Linguistics, pp. 25–32.

Misra, H., Cappé, O. & Yvon, F. (2008), 'Using LDA to detect semantically incoherent documents', *Proc. of CoNLL 2008* pp. 41–48.

Misra, H., Hopfgartner, F., Goyal, A., Punitha, P. & Jose, J. (2010), 'Tv news story segmentation based on semantic coherence and content similarity', *Advances in Multimedia Modeling* pp. 347–357.

Misra, H., Yvon, F., Cappé, O. & Jose, J. (2011), 'Text segmentation: A topic modeling perspective', *Information Processing & Management* **47**(4), 528–544.

Misra, H., Yvon, F., Jose, J. & Cappe, O. (2009), Text segmentation via topic modeling: an analytical study, *in* 'Proceeding of the 18th ACM conference on Information and knowledge management', ACM, pp. 1553–1556.

Nguyen, V., Nguyen, L. & Shimazu, A. (2011), 'Improving text segmentation with non-systematic semantic relation', *Computational Linguistics and Intelligent Text Processing* pp. 304–315.

Passonneau, R. & Litman, D. (1997), 'Discourse segmentation by human and automated means', *Computational Linguistics* **23**(1), 103–139.

Riedl, M. & Biemann, C. (2012), TopicTiling: A Text Segmentation Algorithm based on LDA, *in* 'Student Research Workshop of the 50th Meeting of the Association for Computational Linguistics'.

Sun, Q., Li, R., Luo, D. & Wu, X. (2008), Text segmentation with lda-based fisher kernel, *in* 'Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers', Association for Computational Linguistics, pp. 269–272.

Utiyama, M. & Isahara, H. (2001), A statistical model for domain-independent text segmentation, *in* 'Proceedings of the 39th Annual Meeting on Association for Computational Linguistics', Association for Computational Linguistics, pp. 499–506.

Zhai, Y. & Shah, M. (2006), 'Video scene segmentation using Markov Chain Monte Carlo', *IEEE Transactions on Multimedia* **8**(4), 686–697.