Text Segmentation via Topic Modeling: An Analytical Study

Hemant Misra
Dept. of Computing Science
University of Glasgow
Glasgow, U.K.
hemant@dcs.gla.ac.uk

François Yvon
Univ Paris-Sud 11 &
LIMSI-CNRS
Orsay, France
yvon@limsi.fr

Joemon M. Jose
Dept. of Computing Science
University of Glasgow
Glasgow, U.K.
jj@dcs.gla.ac.uk

Olivier Cappé
TELECOM ParisTech &
LTCI/CNRS
Paris, France
cappe@telecom-paristech.fr

ABSTRACT

In this paper, the task of text segmentation is approached from a topic modeling perspective. We investigate the use of latent Dirichlet allocation (LDA) topic model to segment a text into semantically coherent segments. A major benefit of the proposed approach is that along with the segment boundaries, it outputs the topic distribution associated with each segment. This information is of potential use in applications like segment retrieval and discourse analysis. The new approach outperforms a standard baseline method and yields significantly better performance than most of the available unsupervised methods on a benchmark dataset.

Categories and Subject Descriptors: I.5.4 [**Pattern Recognition**]: Applications - *text processing*

General Terms: Algorithms, Experimentation, Performance

Keywords: text segmentation, unsupervised topic modeling, latent Dirichlet allocation, dynamic programming

1. INTRODUCTION

Text segmentation is the task of dividing a given text data into topically coherent segments [9, 13, 1, 6, 15]. Text segmentation is a fundamental requirement for many IR applications, e.g., segmenting a news broadcast transcription into stories (if possible, with a topic tag) could be very useful for browsing/retrieval. If no text segmentation is performed and a user needs to access a particular story in a news broadcast, he may have to view the entire broadcast to get the story. In contrast, if the news is segmented (either manually or automatically) into stories and labeled, the relevant story can be retrieved directly. Text segmentation can also improve a user's retrieval experience by segmenting a document into topics and subtopics, and presenting only the relevant parts of the document during a search operation. Text segmentation can be useful in tasks such as text summarization and discourse analysis [9].

Several approaches have been proposed in the past to perform this task. Most of the unsupervised approaches exploit *lexical chain* information, the fact that related or similar words tend to be repeated in topically coherent segments and segment boundaries often correspond to a change in the vocabulary [9, 6, 15]. Such approaches do not require a training phase (data), and can be directly applied to any text from any domain, subject to the (only) constraint that word boundaries can be identified. A potential drawback of most of these approaches is that even when the segment boundaries are estimated correctly, the segments are not associated (labeled) with any topic information.

A new approach for text segmentation is proposed in this paper which builds upon well established latent Dirichlet allocation (LDA) [3] model. LDA is a generative and unsupervised topic model; during training it learns the semantics information from the dataset and hence does not rely on mere word repetitions to segment the text. This is a departure from the lexical chain approaches which are typically knowledge-free.

The proposed LDA based approach also differs from the lexical chain approaches in that it "jointly" performs segmentation and topic labeling (outputs the topic distribution associated with each segment). An expected benefit of this approach is its ability to identify the topic of each segment, thus allowing to track topics within a long document or within a collection.

In the proposed LDA based approach for text segmentation, the model is first trained on a large amount of text data and is then used to segment *running texts* it has not seen earlier (text not used for training). This is one of the differences with the approach reported in [14] where the data to be segmented itself is used to train the LDA model, thus making the approach unfit for segmenting running texts. Moreover, the data to be segmented is usually limited; therefore the LDA parameters may not be estimated reliably. This may be the main reason why the performance reported in [14] is not significantly better than that of the basic *Texttiling* method [9].

The rest of the paper is organized as follows: In Section 2,

CIKM'09, November 2–6, 2009, Hong Kong, China.

we recall the principles of dynamic programming (DP) for text segmentation, first reviewing the method proposed by Utiyama and Ishara [15] (one of the most cited baseline approaches to date) and then explaining how to adapt these principles when fragments are scored under the LDA topic model. In Section 3.2, we compare and analyze the performance of the two methods on Choi's benchmarks [6]. Conclusions of this study and the future directions are discussed in Section 4.

2. ALGORITHMICS OF SEGMENTATION

2.1 DP with Probabilistic Scores

Text segmentation can be efficiently implemented with DP techniques [15]. Assuming that text is represented as a linear

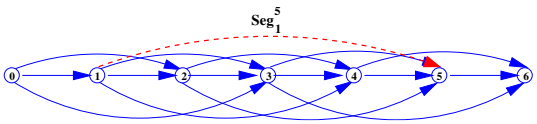

Figure 1: Nodes and segments in dynamic programming.

graph, a segment is defined by two nodes, the begin (B) and the end (E) nodes. For instance, in Fig. 1, segment $\mathbf{Seg_1^5}$ (dotted line) is from begin node B_1 (excluding B_1) to end node E_5 (including E_5). Node 0 is treated as null node for convenience.

In the standard DP approach, scores for all possible node pairs are computed. Therefore, if the graph contains N nodes, one has to consider $N*(N+1)/2$ node pairs.

As described in [15], text segmentation thus proceeds as follows: We denote $d = w_1 w_2 \cdots w_{l_d}$ a document of length l_d ; and $S = S_1 S_2 \cdots S_m$ a particular segmentation S made up of m segments. The likelihood of S is thus

$$P(S|d) = \frac{P(d|S)P(S)}{P(d)} \tag{1}$$

In (1), $P(d|S)$ is the probability of d under segmentation S and $P(S)$ is a prior over segmentations, which corresponds to a penalty factor [15]. Assuming S_i contains n_i word tokens, and that w_i^j denotes the j^{th} word token in S_i, we denote $W_i = w_i^1 \cdots w_i^{n_i}$. Therefore, $d = W_1 \cdots W_m$ with $l_d = \sum_{i=1}^{m} n_i$. Under these assumptions, W_i and S_i are in a one to one correspondence. Further, assuming that segments are independent of each other, (1) can be rewritten as [1]:

$$\begin{aligned} P(S|d) &\propto \left[\prod_{i=1}^{m} P(W_i|S)\right] P(S) \quad \propto \left[\prod_{i=1}^{m} P(W_i|S_i)\right] P(S) \\ &\propto \left[\prod_{i=1}^{m}\prod_{j=1}^{n_i} P(w_i^j|S_i)\right] P(S) \end{aligned} \tag{2}$$

The most likely segmentation is defined as $\hat{S}$=$\underset{S}{\text{argmax}}\ P(S|d)$, and can be recovered using DP in a manner similar to the resolution of shortest path problems. During the forward-pass, for each pair of nodes (B, E), the score of Seg_B^E is computed. The path that maximizes the cumulative score from the first to the last node is searched, and for each E node the value of the best start node B is stored. The information about the best start node is used during trace back to find the path that maximizes the score, and in turn, the segment boundaries.

[1]For a given document d, $P(d)$ is constant for all the segmentations and can be dropped from the equation.

2.2 Scoring Segments by Baseline

The method proposed in [15] consists of modeling each segment using the conventional multinomial model, assuming segment specific parameters are estimated using the usual maximum likelihood estimates with Laplace smoothing. In literature, this approach has often been used as a standard baseline and shown to deliver competitive results on several datasets [15, 7]. The second term intervening in the probability of a segmentation is the penalty factor. In [15], it was optimized to $\log P(S) = -m \log(l_d)$ to yield the best performance.

2.3 Scoring Segments by LDA

LDA is a generative unsupervised topic model [3, 8]. In [3], the authors showed that the model can capture semantic information from a collection of documents. They also investigated the use of LDA for the task of text modeling, text classification and collaborative filtering.

This paper explores the use of topic modeling properties of LDA for the task of text segmentation. Our approach is based on the premise that using a topic model may allow better detection of segment boundaries because segment change should be associated with a significant change in the topic distribution.

LDA adopts the traditional view that texts are represented as word count vectors, and relies upon a two step generation process for these vectors. A key assumption is that *each document is represented by a specific topic distribution* and *each topic has an underlying word distribution*. Gibbs sampling is used in this paper to train the LDA model [8].

In LDA model, the probabilistic generative story of a document is as follows: assuming a fixed and known number of topics, T, for each topic t, a distribution ϕ_t is drawn from a Dirichlet distribution of order V, where V is the vocabulary size of the training corpus. The first step for generating a document is to draw a *topic distribution*, $\Theta = \{\theta_t, t = 1...T\}$ from a Dirichlet distribution over the T-dimensional simplex. Next, assuming that the document length is fixed, for each word occurrence in the document, a topic, z, is chosen from Θ and a word is drawn from the word distribution associated with the topic z. Given the topic distribution, each word is thus drawn independently from every other word using a *document specific* mixture model.

Given Θ, the likelihood of a document, represented as a count vector C, is given by

$$P(C|\Theta, \Phi) = \prod_{v=1}^{V} \left[\sum_{t=1}^{T} (\theta_t \phi_{tv})\right]^{C_v} \tag{3}$$

where C_v is the count of word v in the document.

Being a generative model, LDA can also be used to make predictions regarding novel documents (*assuming they use the same vocabulary as the training corpus - vocabulary mismatch issue in LDA model is explained later*). As the topic distribution of a test document gives its representation along

the latent semantic dimensions, computing this distribution is important in many contexts, including the present task of text segmentation. This computation can be performed using the iterative procedure suggested in [10, 12], which relies on the following update rule

$$\theta_{dt} \leftarrow \frac{1}{l_d} \sum_{v=1}^{V} \frac{C_{dv}\theta_{dt}\phi_{tv}}{\sum_{t'=1}^{T} \theta_{dt'}\phi_{t'v}} \quad (4)$$

where l_d is the document length, computed as the number of running words.

As discussed in [12], this update rule converges monotonically towards a local optimum of the likelihood, and convergence is typically reached in less than $10-15$ iterations. Once the Θ has been obtained for a document, the likelihood of the document can be computed by (3). This recently proposed step for computing Θ for unseen documents is key to computing the likelihood of a document. In this paper, we extend this idea to compute likelihood of a segment and use the estimated likelihood of segments as scores for performing the text segmentation task.

The LDA based method proposed in this paper is based on the following premise: if a segment is made up of only one story it will have only a few active topics, whereas if a segment is made up of more than one story it will have a comparatively higher number of active topics. Extending this reasoning further, if a segment is coherent (the topic distribution for a segment has only a few active topics), the log-likelihood for that segment is typically high, as compared to the log-likelihood in the case when a segment is not coherent [12]. This observation is of critical importance in the success of the proposed LDA based approach for text segmentation task, and has been left unexplored except for its original use in detecting coherence of a document [12]. It is thus tempting to use the log-likelihood of each possible segment as a score in the DP algorithm and to recover the segmentation from the path that yields the highest log-likelihood.

The proposed LDA based approach for text segmentation task works like this:

1. For each possible segment, S_i,
   (a) Compute its Θ by performing 15 iterations of (4): $\theta_{S_i t} \leftarrow \frac{1}{n_i} \sum_{v=1}^{V} \frac{C_{W_i v}\theta_{S_i t}\phi_{tv}}{\sum_{t'=1}^{T} \theta_{S_i t'}\phi_{t'v}}$
   (b) Compute its log-likelihood using (3): $P(W_i|\Theta,\Phi) = \prod_{v=1}^{V} \left[\sum_{t=1}^{T}(\theta_{S_i t}\phi_{tv})\right]^{C_{W_i v}}$
   (c) The likelihood of the segment is treated as its score: $\log P(W_i|S_i) = \log P(W_i|\Theta,\Phi)$
2. Substitute the scores of the segments in (2), and use DP to find the segmentation which maximizes the score.

The penalty factor we used is defined as $\log P(S) = -p \cdot m \cdot \log(l_d)$, where $p = 3$ was empirically found to yield the best performance on some heldout dataset and used throughout.

3. EXPERIMENTAL RESULTS

3.1 Databases

The dataset used in this study is the Choi's dataset (`http://www.freddychoi.co.uk/`), which has been used repeatedly in benchmarking text segmentation algorithms. Choi's dataset is derived from Brown corpus which consists of running text of edited English prose printed in US during the calendar year 1961. Choi's dataset is divided into 4 subsets ("3-5", "6-8", "9-11" and "3-11") depending upon the number of sentences in a segment/story. For example, in subset "X-Y", a segment is derived by (randomly) choosing a story from Brown corpus, followed by selecting first N (a random number between X and Y) sentences from that story. Exactly 10 such segments are concatenated to make a document. Further, in each subset there are 100 documents to be segmented. By design, the segments are not complete stories.

From Reuters Corpus Volume 1 (RCV1) [11] collection, we selected 27,672 news items for training the LDA model (*ReutersTrain*). In these experiments, the number of topics (T) and Dirichlet priors (α and β) are set to the following values: $T = 50$, $\alpha = 1$ and $\beta = 0.01$. A standard practice in the task of text segmentation is to assume that sentence boundaries are known [6, 5, 15]. We also make use of this information, that is, each sentence beginning is a possible B node and each sentence end is a possible E node.

3.2 Results and Analysis

In this section, we compare the results of the following segmentation systems (Table 1): (i) The results reported in [15] (using Choi's implementation of Porter stemmer), (ii) Our own implementation of Utiyama's method, and (iii) LDA based segmentation.

Method	Porter Stemmer	P_k, in % (time, in sec)			
		3-5	6-8	9-11	3-11
Baseline	Choi's	13	6	6	11
Baseline	None	14	7	7	11
LDA	None	22.5	15.4	13.1	15.5

Table 1: ***The performance of text segmentation algorithms on Choi's dataset.***

The results are presented in terms of P_k, the probabilistic error metrics introduced in [1]. P_k is the probability that two randomly drawn sentences which are k sentences apart are classified incorrectly. As in [6], k is set to the average segment length in our experiments. A lower value of P_k indicates a higher accuracy in text segmentation.

The results reported in Table 1 suggest that the performance of both the approaches improves with an increase in the segment size. This is an expected result: longer segments allow a better estimation of the multinomial parameters for the baseline method and of the topic distribution for LDA.

Compared to LDA based approach, the baseline method is more accurate. An inspection of outputs gives a possible explanation for this: There is a serious mismatch in vocabulary between *ReutersTrain* dataset (used for LDA training) and Choi's dataset used for testing. Similar issues of semantic mismatch were highlighted, for example, in [2], where the authors used a generic latent semantic space while performing text segmentation. The baseline method utilizes the full available vocabulary and all the content words (except the stop words that were removed before segmenting the text) for computing the score of a segment. In contrast, the vocabulary of an LDA model is defined by its training data. During segmentation, the content words in the data (text to be segmented) that are not present in the train-

ing vocabulary are not used for computing the score of a segment. Comparing the baseline and LDA methods, approximate loss in vocabulary and content words by LDA is 11.6% and 10.5% respectively.

To overcome this problem of vocabulary mismatch, we divided the Choi's dataset into two parts. The first 50 documents from each subset were put in SET A (50 documents * 10 segments/document * 4 sub-sets = 2000 segments) and the last 50 documents from each subset were put in SET B. SET A was used along with ReutersTrain to train the LDA model and SET B was used for testing. The results for the baseline and adapted LDA models for SET B are given in Table 2. The results present in Table 2 show that

Method	P_k, in %			
	3-5	6-8	9-11	3-11
Baseline	14.9	8.1	7.7	11.2
Unadapted LDA	23.0	15.8	14.4	16.1
Adapted LDA	**2.2**	**2.3**	**4.1**	**2.3**
Choi et al [6]	12	9	9	12
Utiyama and Isahara [15]	9	7	5	10
Choi et al [5]	10	7	5	9
Brants et al [4]	7.4	8.0	6.8	10.7
Fragkou et al [7]	**5.5**	**3**	**1.3**	**7**

Table 2: ***The text segmentation performance on the SET B of Choi's dataset by the baseline, Unadapted LDA (ReutersTrain) and Adapted LDA (ReutersTrain+Choi SET A) methods. The performance by several other methods for complete Choi's dataset is also mentioned from their respective papers.***

the vocabulary mismatch was the main reason for the poor performance of LDA based method. The performance of the adapted LDA model is significantly better than that of the unadapted LDA model, and it also performs better than the baseline method (improvement is statistically significant with respect to the baseline). In fact, the performance of the adapted LDA model is better than any other unsupervised model previously reported in the literature for the text segmentation task on this benchmark [6, 15, 5, 7]. Noticeably, all these results are for the case when the algorithms used the information about the number of segments for getting the best performance, and in certain cases some data was used for adaptation [15, 7]. In the table, [7] has better performance than adapted LDA model for subset "9-11". This particular method introduced some extra variables in the DP algorithm to capture the local as well as global dynamics. These variables were first adapted over a train dataset and then the best performance was reported. Lastly, though our baseline method is also from [15], it does not utilize the information about the number of segments. That is why the results of first and fifth rows are different in Table 2.

4. CONCLUSIONS

In this study, we proposed an LDA based method for the text segmentation task. The proposed method computes topic distributions jointly with segmentation, thus allowing to collect information about the thematic content of each segment. This information can be used to keep track of recurring topics.

We investigated and compared the performance of our method with a standard approach often used as a baseline for the text segmentation task [15]), and analyzed its potential strengths and weaknesses. The LDA based method gave a better performance than that of the baseline in adapted conditions (a small amount of data from the test domain is used along with a large amount out-of-domain data to train the LDA model). In fact, the proposed LDA based method outperformed every other unsupervised approach on Choi's dataset.

5. ACKNOWLEDGMENTS

This research was supported by the European Commission under the contracts *FP6-045032-SEMEDIA*, *FP6-033715-MIAUCE* and *FP6-027122-SALERO*.

6. REFERENCES

[1] D. Beeferman, A. Berger, and J. Lafferty. Statistical models for text segmentation. *Machine Learning*, 34(1–3):177–210, 1999.

[2] Y. Bestgen. Improving text segmentation using latent semantic analysis: A reanalysis of Choi, Wiemer-Hastings, and Moore (2001). *Computational Linguistics*, 32(1):5–12, 2006.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems (NIPS)*, volume 14, pages 601–608, Cambridge, MA, 2002. MIT Press.

[4] T. Brants, F. Chen, and I. Tsochantaridis. Topic-based document segmentation with probabilistic latent semantic analysis. In *Proceedings of the International Conference on Information and Knowledge Management*, pages 211–218, McLean, Virginia, U.S.A., 2002. ACM.

[5] F. Choi, P. Wiemer-Hastings, and J. Moore. Latent semantic analysis for text segmentation. In *Proceedings of EMNLP*, pages 109–117, Pittsburgh, PA, U.S.A., 2001.

[6] F. Y. Y. Choi. Advances in domain independant linear text segmentation. In *Proceedings of the Conference of North American Chapter of the ACL*, pages 26–33, Seattle, WA, U.S.A., 2000.

[7] P. Fragkou, V. Petridis, and A. Kehagias. A dynamic programming algorithm for linear text segmentation. *Journal of Intelligent Information System*, 23(2):179–197, 2004.

[8] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101 (supl 1):5228–5235, 2004.

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

[10] A. Heidel, H. an Chang, and L. shan Lee. Language model adaptation using latent Dirichlet allocation and an efficient topic inference algorithm. In *Proceedings of EuroSpeech*, Antwerp, Belgium, 2007.

[11] D. D. Lewis, Y. Yang, T. Rose, and F. Li. RCV1: A new benchmark collection for text categorization research. *Machine Learning Research*, 5:361–397, 2004.

[12] H. Misra, O. Cappé, and F. Yvon. Using LDA to detect semantically incoherent documents. In *Proceedings of CoNLL*, pages 41–48, Manchester, U.K., 2008.

[13] J. C. Reynar. *Topic Segmentation: Algorithms and Applications*. PhD thesis, University of Pennsylvania, 1998.

[14] Q. Sun, R. Li, D. Luo, and S. Wu. Text segmentation with LDA-based Fisher kernel. In *Proceedings of ACL-08: HLT, Short Papers*, pages 269–272, Columbus, Ohio, June 2008.

[15] M. Utiyama and H. Isahara. A statistical model for domain-independent text segmentation. In *Meeting of the Association for Computational Linguistics*, pages 491–498, Bergen, Norway, 2001.