Domain-independent Text Segmentation Using Anisotropic Diffusion and Dynamic Programming

Xiang Ji Hongyuan Zha
Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA
{xji, zha}@cse.psu.edu

ABSTRACT

This paper presents a novel domain-independent text segmentation method, which identifies the boundaries of topic changes in long text documents and/or text streams. The method consists of three components: As a preprocessing step, we eliminate the *document-dependent* stop words as well as the generic stop words before the sentence similarity is computed. This step assists in the discrimination of the sentence semantic information. Then the cohesion information of sentences in a document or a text stream is captured with a sentence-distance matrix with each entry corresponding to the similarity between a sentence pair. The distance matrix can be represented with a gray-scale image. Thus, a text segmentation problem is converted into an image segmentation problem. We apply the anisotropic diffusion technique to the image representation of the distance matrix to enhance the semantic cohesion of sentence topical groups as well as sharpen topical boundaries. At last, the dynamic programming technique is adapted to find the optimal topical boundaries and provide a zoom-in and zoom-out mechanism for topics access by segmenting text in variable numbers of sentence topical groups. Our approach involves no domain-specific training, and it can be applied to texts in a variety of domains. The experimental results show that our approach is effective in text segmentation and outperforms several state-of-the-art methods.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval; I.2.7 [**Computing Methodologies**]: Natural Language Processing

General Terms

Algorithms, Experimentation, Theory

SIGIR'03, July 28–August 1, 2003, Toronto, Canada.

Keywords

text segmentation, document-dependent stop words, anisotropic diffusion, dynamic programming

1. INTRODUCTION

The task of text segmentation is to determine the boundaries between topics in long text documents or text streams and divide the texts into a set of text segments, each of which consists of a consecutive sequence of sentences or paragraphs sharing a coherent topic. Several important applications in information retrieval and text mining make use of text segmentation as an important enabling technique, and we briefly list a few in the following: In traditional information access of text databases, we assume a notion of text documents, and then attempt to search and retrieve documents that satisfy users information needs. However, the increasing lengths of documents in full-text collection motivates smaller and more coherent passage level retrieval, which has been demonstrated to improve retrieval performance as well as reduce information overload in processing and transmission [6, 24]. In addition, some text collections, such as newswire feeds and streams of automatic speech recognition transcripts, do not have an explicit notion of documents. These text streams usually contain many distinct topics with no explicitly marked boundaries between the topics. Human can usually recognize the topical boundaries without much difficulty. However, with the increase in the sizes of text streams collection, it is infeasible to have human readers to manually mark the boundaries. Thus, automatic text segmentation based on topics becomes a critical task for developing effective and efficient systems to access those collections of text streams. Last but not least, the task of text segmentation is also utilized in some of the approaches for text summarization. Since long text documents often discuss multiple topics, revealing the latent topical structures of textual documents enables summarizing subtopics of documents instead of the whole documents [14, 29]. The ability of segmenting texts at various levels of granularity helps to generate summaries with different compression rates and coverage diversities [17, 28].

Text segmentation tasks in general fall into two categories: The first type is to identify the locations of topic changes for text streams. Text streams, such as streams of transcripts from automatic speech recognition, newswire feeds, and television closed-caption transcripts, usually contain noticeable topic transitions. Text segmentation for text streams targets those transition points in the text streams. The second type of text segmentation is to identify and isolate subtopics by splitting substantive long documents. The long documents usually address several closely related subtopics or several aspects of the same topic. Because the topic transitions are much subtler than those in text streams, segmenting long documents, which is less well addressed in previous research work, tends to be a more challenging task than segmenting text streams.

In this paper, we explore the idea of converting a text segmentation problem into an image segmentation problem. We employ the anisotropic diffusion technique, a technique that has been successfully used for deblurring noisy images in image processing [18], and develop a novel approach for both text streams with noticeable topic transitions and documents with subtle subtopic shifts. Our approach, which is unsupervised and domain-independent, can be applied to texts in many application fields. Instead of utilizing cue-words that indicate topic transition, we rely on lexical cohesion information of texts for topic boundaries detection which tends to alleviate the performance deterioration caused by possible errors in transcripts from speech recognition. The major contributions of this paper are as follows: We capture the sentence cohesion information of a document or a text stream with a square sentence dissimilarity matrix, each entry corresponding to the distance of a pair of sentences. The cohesion information can then be represented by a gray scale image, which enables converting the text segmentation problem to an image segmentation problem. We use the anisotropic diffusion method to simultaneously enhance the lexical cohesion of sentences within a topical group and sharpen the boundaries of the topic transitions. We also propose the concept of document-dependent stop words as an effective tool to reduce the correlation among the topical groups in long documents. Lastly, the dynamic programming technique is adapted to segment texts in variable levels of topical granularity, which is much more flexible than those used in previous research.

This paper is divided into five sections. We review previous work on various text segmentation methods in Section 2. Section 3 presents our proposed method, which contains the sentence similarity measure, distance matrix construction, document-dependent stop words computation, application of anisotropic diffusion method, and the customized dynamic programming technique. In Section 4, we discuss details of our experiments. We conclude the paper in Section 5 and discuss several possible future research topics.

2. PREVIOUS WORK

There have been several research done on text segmentation especially those in connection with the topic detection and tracking (TDT) research [1, 26]. Most of previous work is targeted at topic tracking of broadcast speech data and newswire texts. Neither their algorithms nor their experimental data sets deal with subtopic segmentation of substantive long documents. Those existing approaches in general fall into two categories: supervised methods [2, 4, 5, 8, 22, 25, 27] and unsupervised methods [7, 11, 12, 15, 16].

Those supervised learning approaches achieve good performance based on large amount of training data in some specific domains. However, the systems developed are not guaranteed to perform well when they are applied to a different text domain without adequate amount of training data. On the contrary, the unsupervised methods tend to be more domain-independent. Based on information used in segmentation, text segmentation methods can be put into two groups: some relying on lexical cohesion and others utilizing multi-source information. The essential assumption of lexical cohesion-based methods is that similar vocabulary tends to be in a part of a coherent topic segment. The lexical cohesion-based methods utilize word repetition [19, 15, 22], context vectors [10], and semantic similarity [2, 7, 12] to identify cohesion. Multi-source based methods employ not only lexical cohesion but also indicators of topic transition in presentation format and regular cues [8]. These cues can improve accuracy for topic segmentation tasks. However, usually there are high error rates in the transcripts from speech recognition and television closed-caption transcripts. These errors make it difficult for cue recognition, and thus deteriorate the performance of text segmentation algorithms based on cue recognition in real-world applications [4].

Dynamic programming technique, in particular, has been applied to text segmentation [12, 19], where a length model is required in order to achieve a reasonable performance. However, this model is not applicable when the size and number of text segments vary significantly. We adapt the dynamic programming technique without requiring the text segmentation length model. The optimal number of segments and segmentation boundaries are automatically determined by our algorithm. In addition, users are able to set the number of segments for a document or a text stream, and then our method will generate the optimal segmentation with the number of segments defined by the user. By this means, users are able to access (sub)topics of a text document at different granularity.

Several experimental studies have been reported for text segmentation. Currently, the best performance is achieved by Brants et.al. [5] and Utiyama et.al. [25] based on the Brown data sets. However, their experiments are limited to a set of artificially generated text documents concatenating texts with different contents or topics. The effectiveness of their algorithms in segmenting coherent documents for subtopics is unknown.

3. METHODOLOGY

For our purpose, a text segment consists of a consecutive sequence of sentences. However, other text units such as paragraphs or text lines can also be used to substitute for sentences. The goal of text segmentation is to identify the boundaries of the text segments in a document or a text stream. First, sentences are tokenized into words, the words are stemmed, and generic stop words and document-dependent stop words, if necessary, are removed. After the preprocessing, sentences are represented by word-frequency vectors, based on which the pairwise distances are computed, and the sentence distance matrix are formed. In general, the distance matrix tends to capture the sentence cohesion information in a document or a text stream. The distance matrix is further processed by a technique — anisotropic diffusion — used in image processing to sharpen the segment boundaries. At last, we apply dynamic programming technique to the processed distance matrix to find the optimal segmentation boundaries.

3.1 Sentence Distance Matrix

A document with m sentences is modelled with a set of sentence vectors $S = \{s_1, \dots, s_m\}$, and each s_i corresponds to a sentence. The entries of the sentence vectors correspond to words occurring in the sentence and take values of the corresponding word frequencies. Stop words and document-dependent stop words, which will be discussed in the next subsection, are removed before building the sentence vectors. The distance d_{ij} between the sentence pair s_i and s_j is calculates as

$$d_{ij} = 1 - \frac{s_i^T s_j}{\|s_i\|\|s_j\|},$$

where $\|\cdot\|$ denotes the Euclidean length of a vector. The sentence distance matrix of a document, $D = [d_{ij}]_{m\times m}$, is generated by computing all pairwise distances of the sentences in the document. Obviously, the distance matrix is a symmetric square matrix because $d_ij = d_ji$. For example, the sentence distance matrix for a document with three subtopics is displayed as a gray-scale image in Figure 1. In the image, sentences are arranged from left-to-right/top-to-down in the same order as they are in the document. Each pixel in the image corresponds to the distance between the corresponding pair of sentences. Brighter pixel means larger distance between the pair of sentences. The pixels along the diagonal from top-left to bottom-right are black, which means a sentence has zero-distance from itself. The text segmentation task is to find a sequence of adjacent sub-matrices along the top-left to bottom-right diagonal satisfying certain criteria. Each sub-matrix corresponds to a consecutive sequence of sentences in the original text.

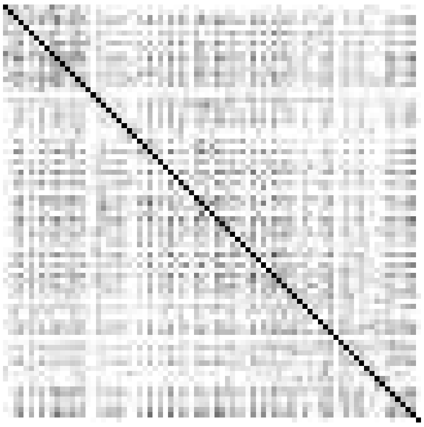

Figure 1: Sentence distance matrix before document-dependent stop words removed

3.2 Document-dependent Stop words

Generic stop words consisting of function words such as conjunctions, propositions, pronouns, etc. are usually removed when constructing the sentence vectors. For text segmentation of a long documents, there is another type of words which we call *document-dependent* stop words that should also be removed. Those are the kind of words that are useful in discriminating among several different documents but are rather harmful in detecting subtopics in a document. As an example, consider segmenting a long article about heart diseases which might include several subtopics such as anatomy and physiology, clinical trials, diagnosis and symptoms, and rehabilitation and treatment. The frequency of the word *heart* is certainly very high in this document, and it is a very useful word for identifying the document as one about heart diseases. However, the word *heart* also tends to appear in each of the subtopics with high frequencies and in general will not serve well to distinguish the subtopic boundaries. Removing the word *heart* can break the correlation among the different subtopics and hence stress their differences. One thing we need to be careful about identifying words as document-dependent stop words is that relative high frequency alone is *not* sufficient for labling a word as document-dependent stop word. The word are also required to be uniformly distributed in the whole document. For example, the word *surgery* can also appear at very high frequency in the document, but we can not characterize it as a document-dependent stop word because it almost exclusively occurs in the subtopic about heart disease treatment.

Based on the above discussion, we propose the following steps to select document-dependent stop words.

1. Count all words' frequencies in a document, and select those words with top $k\%$ highest frequencies as candidate document-dependent stop words;

2. Divide a document with m sentences *evenly* into i segments for $i = 2, \dots, l$. For a candidate document-dependent stop word w, count w's frequencies in each of the i segments as c_{ij}, $1 \le j \le i, i = 2, \dots, l$;

3. Define $\widehat{m_i} = \sum_{j=1}^{i} c_{ij}/i$. Compute the variance of the word's frequency for each segmentation as

$$var_i = (\sum_{j=1}^{i}(c_{ij} - \widehat{m}_i)^2)/(i-1);$$

4. Compute the average variance across the l segmentations as $\widehat{var} = \sum_{i=2}^{l} var_i/(l-1)$. If $\widehat{var}$ is greater than a certain threshold ϵ, then the candidate stop word is a document-dependent stop word.

The parameters k, l and ϵ need to be chosen in order to apply the above selection procedure. Ideally they should be chosen using methods such as cross-validation to optimize certain performance criterion for text segmentation. We empirically choose them as $k = 5$, $l = m/4$, and $e = 0.6$.

After removing the generic stop words and document-dependent stop words, we apply the Porter stemming algorithm [20] to the remaining words to obtain word stems, and each sentence vectors are indexed by the resulted word stems. Figure 2 illustrates the sentence distance matrix after the two types of stop words are deleted. There are fewer dark-square regions outside of the main diagonal area than those in the Figure 1, which makes the dark-square regions along the main diagonal of the distance matrix more outstanding. However, the boundaries are still relatively blurry. In the next section, we will discuss a method to enhance the contrast among different topical groups represented as square submatrices along the main diagonal of the distance matrix.

3.3 Anisotropic Diffusion

A common assumption in text segmentation research is that lexical information of sentences tend to be consistent with their semantic information. Thus, researchers may rely on the lexical information, such as word repetition, textual

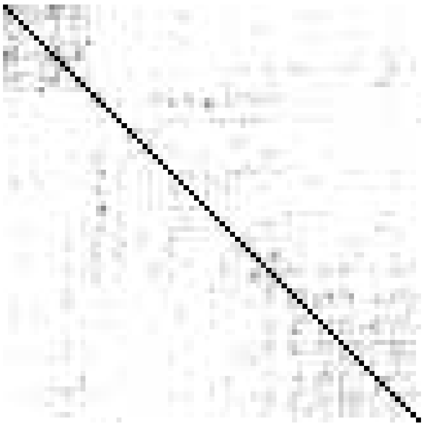

Figure 2: Sentence distance matrix after document-dependent stop words removed

substitution, co-reference, and conjunction, to recognize the coherent topic segments. Because of the abundance and variation of vocabulary and complexity of thoughts expression, fully extracting and utilizing semantic information of text is a very challenging task through lexical information. It has been demonstrated that lexical repetition alone can be used to determine the degree of cohesion between pair of sentences in some circumstance [3]. However, people sometimes fail to describe a theme with correct vocabulary or cannot arrange lexical information into a coherent semantic substance for some reasons. Thus, it is not unusual that lexical information does not effectively convey intended semantic information. When lexical information is captured with distance matrix, ideally, there should be a sequence of dark-square regions along the main diagonal of the distance matrix, each corresponding to a sentence topical group. However, because of reasons listed above, certain amounts of noise do appear in these dark-square regions, and their existence tend to blur the boundaries of the square regions, just as is shown in Figure 2. Instead of involving more complex natural language processing techniques for deeper linguistic analysis, we resort to applying a technique from image processing — anisotropic diffusion — to the sentence distance matrix to deblur the noise inside each dark-square region while at the same time to sharpen the boundaries of dark-square regions.

Anisotropic diffusion was proposed by Perona and Malik [18] as a powerful tool for image enhancement. Its goal is to reduce noise in homogeneous regions of an image, making homogeneous regions even more homogeneous, while at the same time also sharpen boundaries between homogeneous regions. Thus, the anisotropic diffusion keeps a strong flow within homogeneous regions, and little flow is allowed across region boundaries. In order to achieve this property, the diffusion coefficient function is chosen to be inversely proportional to the size of the gradient so that the diffusion flow increases within homogeneous regions where the gradient tends to be small. The following two diffusion coefficient functions have been proposed in [18]:

$$g(\nabla I) = e^{(-(\|\nabla I\|/K)^2)}, \tag{1}$$

and

$$g(\nabla I) = \frac{1}{1 + (\frac{\|\nabla I\|}{K})^2}, \tag{2}$$

where ∇ denote the gradient operator. Let $I^t_{i,j}$ denote the intensity at pixel (i, j), its evolution is then governed by the following equations:

$$I^{t+1}_{i,j} = I^t_{i,j} + \lambda[c_N \cdot \bigtriangledown_N I + c_s \cdot \bigtriangledown_s I + c_E \cdot \bigtriangledown_E I + c_W \cdot \bigtriangledown_W I]^t_{i,j},$$

where $0 \le \lambda \le 0.25$, N, S, E, W are the mnemonic subscripts for *North*, *South*, *East*, *West*, and $\bigtriangledown$ indicates nearest-neighbor differences. For each pixel in the image, we can calculate $\bigtriangledown I_{i,j}$ in each direction with:

$$\bigtriangledown_N I_{i,j} = I_{i-1,j} - I_{i,j}$$

$$\bigtriangledown_S I_{i,j} = I_{i+1,j} - I_{i,j}$$

$$\bigtriangledown_E I_{i,j} = I_{i,j+1} - I_{i,j}$$

$$\bigtriangledown_W I_{i,j} = I_{i,j-1} - I_{i,j}$$

At the same time the c_{ij} in four directions can be calculated with:

$$c_{i,j} = g(\|\nabla I_{i,j}\|),$$

where function g can be either (1) or (2). The first function favors high-contrast edges over low-contrast edges, while the second function favors wide regions over smaller ones.

We apply the anisotropic diffusion technique to sentence distance matrices. Each value of the distance matrix corresponds to a pixel of the image. Text segments correspond to homogeneous regions in the distance matrix, and there are boundaries separating these homogeneous regions. We use the equation (2) in our experiments. With diffusion over time, the homogeneity within each region increases as small grey level variations are smoothed out. At the same time, the boundaries between regions become more pronounced. We also quantize the distance matrix D after the anisotropic diffusion for better boundaries display with:

$$\hat{d}_{ij} = \begin{cases} 0 & \text{if } 0.1 \le d_{ij} \le 0.95 \\ 1 & \text{otherwise} \end{cases}$$

The gray scale image in Figure 3 illustrates the distance-matrix after the anisotropic diffusion process and quantization. Now, we can recognize that there are three sentence topical groups in the sample documents, which we know in advance.

3.4 Segmentation by Dynamic Programming

Within the context of the sentence distance matrix, text segmentation amounts to partition the matrix into K blocks of sub-matrix along the diagonal. Specifically, we denote $\hat{D} = [\hat{d}_{ij}]_{m \times m}$ as the distance matrix after applying anisotropic diffusion. We aim to partition $\hat{D}$ into $(\hat{D}_{ij})^K_{i,j=1}$. Each $\hat{D}_{ij}$ is a square sub-matrix and corresponds to a sentence topical group including sentence $s_i, s_{i+1}, \ldots, s_j$. There is a $e[P(m, K)]$ associated with the text segmentation $(\hat{D}_{ij})^K_{i,j=1}$ with

$$e[P(m, K)] = \sum_{i=1}^{K} Diam(\hat{D}_{ij}),$$

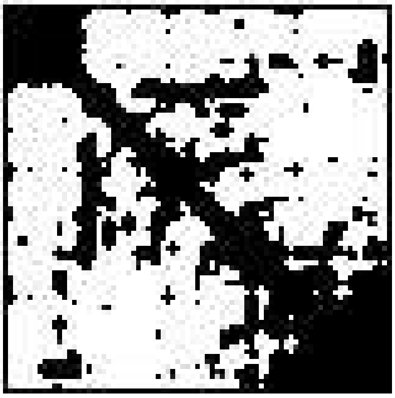

Figure 3: Sentence distance matrix after document-dependent stop words removed and anisotropic diffusion

where $Diam(\hat{D}_{ij})$ is the sum of all elements of $\hat{D}_{ij}$. We apply dynamic programming to find the segmentation $(\hat{D}_{ij})_{i,j=1}^{K}$ with the minimum of $e[P(m, K)]$, the computational steps are listed below [9]:

1. Compute the diameter $Diam(s_I, s_J)$, for all pair of sentences s_I, s_J such that $1 \leq s_I < s_J \leq m$ by $Diam(s_I, s_J) = \sum_{i=s_I}^{s_J} \sum_{j=s_I}^{s_J} \hat{d}_{ij}$;

2. Compute the errors of the optimal partitions, $2 \leq s_I \leq m$, by $e[P(s_I, 2)] = min[Diam(1, s_J - 1) + Diam(s_J, s_I)]$ over the range $2 \leq s_J \leq s_I$;

3. For each $L(3 \leq L \leq m)$ calculate the errors of the optimal partitions $e[P(s_I, L)](L \leq s_I \leq m)$ by $e[P(s_I, L)] = \min\{e[P(s_J - 1, L - 1)] + Diam(s_J, s_I)\}$;

4. In order to decide the optimal number of text segments, calculate the decrease of $e[P(m, i)]$ with $g_i = e[P(m, i)] - e[P(m, i + 1)]$, $2 \leq i \leq m$. The K that $max(g_K/g_{K+1})$ is the optimal number of text segments.

5. The optimal partition $P(m, K)$ is discovered from the table of errors $e[P(I, L)](1 \leq L \leq K, 1 \leq I \leq m)$ by first finding J so that $e[P(m, K)] = e[P(J - 1, K - 1)] + Diam(J, m)$.

6. Based on above calculation, we choose the $(J, J + 1, \ldots, m)$ as the last segment. Then find J' that $e[P(J - 1, K - 1)] = e[P(J - 1, K - 2)] + Diam(J', J - 1)$, so that the $(J', J' + 1, \ldots, J - 1)$ is the second-to-the last segment, and so on.

With a fixed K, dynamic programming applied to the objective function $e[P(m, K)]$ can be proved to achieve a global minimum. The is mainly because $e[P(m, K)]$ can be written as a summation. The criteria in Step 4 also provides a way to select the number of segments K to use. The segments boundaries that correspond to the given number of segments for the text can be retrieved in the same way. Even the number of segments varies, the corresponding segmentations are guaranteed to be the optimal at the given number of segments. This provides the function of zoom-in and zoom-out to access subtopics with different granularity.

Range of n	3-11	3-5	6-8	9-11
Number of samples	400	100	100	100

Table 1: The first data set statistics (Choi, 2000)

Source of Text	Sect1,Ch1	Sect2,Ch2	Sect1,Ch4
# of words	1747	3453	2751
# of paragraphs	8	25	20

Table 2: The second data set statistics

4. EXPERIMENTS

We carried out several experiments with various data sets and parameter settings. This section will discuss some of our experimental results and compare with those done in previous research. We also study the gains in performance of the anisotropic diffusion technique and the use of document-dependent stop words. The experimental results are evaluated with two well-established metrics. In addition, the parameters of anisotropic diffusion procedure vary in our experiments to study their impacts on the final results.

4.1 Test Data

As we mentioned in the beginning of the paper, the tasks of text segmentation fall into two categories: text stream segmentation and coherent document segmentation. The (sub)topics transition and changes are relatively obvious in text stream case, while they are usually subtler and more difficult to detect in coherent documents. In our experiments, we used two data sets.

The first is a synthetic data set that was used in previous research by Choi [7] and Utiyama [25] to study the first type of segmentation task. The same data set has also been tested for other segmentation algorithms including $C99$ [7], TextTiling [11], DotPlot [22], Segmenter [15], and $U00$ [25]. The data set contains 700 samples, each is a concatenation of ten text segments. A segment is the first n sentences of a randomly selected document from the Brown corpus. Table 1 lists the data statistics on n. Since different documents are on different topics, the ten set of sentences in a concatenated text conveys different topics.

The second data set are texts selected from $Mars$ written by Percival Lowell in 1895. We present the results with Section 1 ($As\ a\ Star$) of Chapter 1 ($General\ Characteristics$), Section 2 ($Clouds$) of Chapter 2 ($Atmosphere$), and Section 1 ($First\ Appearances$) of Chapter 4 ($Canals$). Table 2 lists the data statistics. Heinonen also used some texts from the same book to evaluate his text segmentation method in [12].

4.2 Evaluation

We use the error metric proposed by Beeferman et al. [2] to evaluate text segmentation methods. Beeferman et al. extensively discussed other metrics including those in [10, 22, 19]. This criteria has been widely used in [2, 4, 7, 25] to evaluate several text segmentation algorithms. It calculates $P_k(ref, hyp)$, which is the probability of a randomly chosen pair of words a distance of k words apart inconsistently classified. The probability contains the $miss$ and $false\ alarm$ $probabilities$ [2]:

$$p(error|ref, hyp, k) =$$
$$p(miss|ref, hyp, diff, k)p(diff|ref, k)+$$
$$p(falsealarm|ref, hyp, same, k)p(same|ref, k)$$

	3-11	3-5	6-8	9-11
TextTiling	46%	44%	43%	48
Segmenter	36%	23%	33%	43%
Dotplot	22%	21%	18%	16%
C99	13%	18%	10%	10%
U00	11%	13%	6%	6%
TopSeg02	10.74%	7.44%	7.95%	6.75%
Our method	6.0%	7.1%	5.3%	4.3%

Table 3: Error rates on the first data set achieved by algorithms with the numbers of segments unknown in advance

	3-11	3-5	6-8	9-11
C99'	12%	11%	10%	9%
U00'	10%	9%	7%	5%
Our method	6.0%	6.8%	5.2%	4.3%

Table 4: Error rates on the first data set achieved by algorithms with the numbers of segments known in advance

Low error probability means high accuracy.

Recently Pevzner and Hearst pointed out that the above evaluation metric penalizes false negatives more heavily than false positives, over-penalizes near-misses. They proposed another evaluation metric [21]:

$$windowDiff(ref, hyp) = \frac{1}{N-k}\sum_{i=1}^{N-k}(|b(ref_i, ref_{i+k}) - b(hyp_i, hyp_{i+k})|).$$

This evaluation metric is used in [5]. We also present our result based on this metric for comparison as well as future research study.

4.3 Experimental Results

Table 3 shows results on the first data set when the number of segments are determined by the systems, instead of given in advance. They are the best error rates achieved by TextTiling, Segmenter, Dotplot, C99, U00 [7, 25], TopSeg02 [5] and our method. Our approach outperforms other methods in segmenting all of the four subsets of data.

Table 3 shows the results on the first data set when the number of segments are given in advance. They are the best error rates achieved by C99, U00, and our method. Both C99 and U00 reduce the average error rates a lot when the number of segments are given. Our method generates almost the same error rates. The results indicate the adapted dynamic programming technique in our method is very effective and stable in finding the optimal number of segments. Combining with the results in Table 2, it is indicated that our method is more favorable for dealing with long text segments. The smallest error rate is achieved when the average segment size is the largest among all subsets of data.

In order to demonstrate the impact of anisotropic diffusion technique in improving text cohesion in abstract data level, we compare the error rates of segmenting texts with anisotropic diffusion and without anisotropic diffusion employed. At the same time, the experiments are done with the number of segments known and unknown, respectively. Figure 4 shows the error rates in different configuration with bars. There are four groups of bars along x-axis each containing four bars. Each group of bars corresponds to one subset of data from the first data set. On average, the error rates with anisotropic diffusion are reduced by as much as 70% over those without anisotropic diffusion.

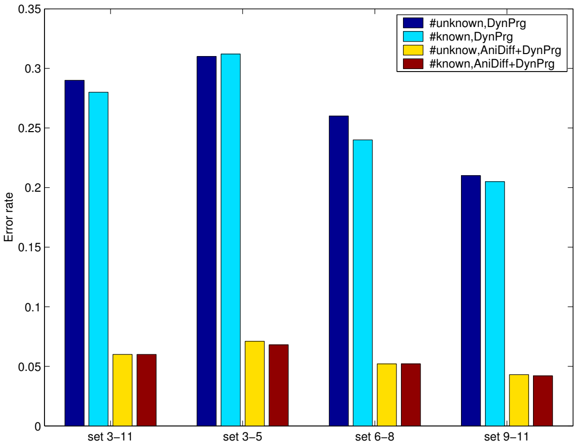

Figure 4: Error rates on the first data set achieved with different configuration of our method

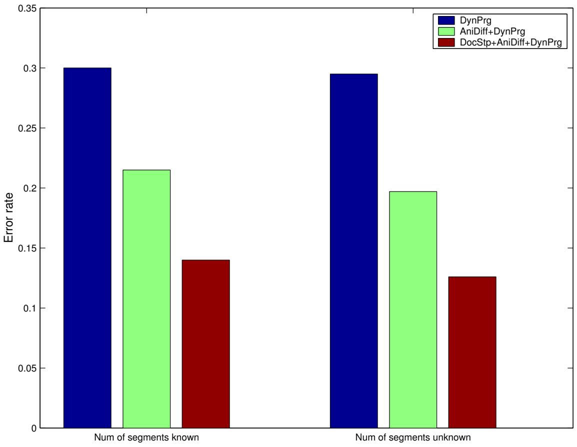

Figure 5: Error rates on the second data set achieved with different configuration of our method

Figure 5 illustrates the experimental results for the second data set. They are the best error rates achieved by our method. Since no previous research extensively discuss the issue, we do not have other's experimental results for comparison. We change the conditions by adding document-stop words removal and anisotropic diffusion. All error rates are plotted in the figure for comparison. There are two groups of bars along x-axis each containing three bars. The first group of bars represents error rates when the segment numbers are not known in advance while the second group of bars represents error rates when the segment numbers are known in advance. The error rates with number of segments known in advance improve about 10% over those without the number of segments. The error rates with document-dependent stop words removed improve about 16% over those with anisotropic diffusion, and improve about 30% over those with dynamic programming only. From the results, we find that the content-dependent stop words are very important in discriminating semantic groups of sentences inside a doc-

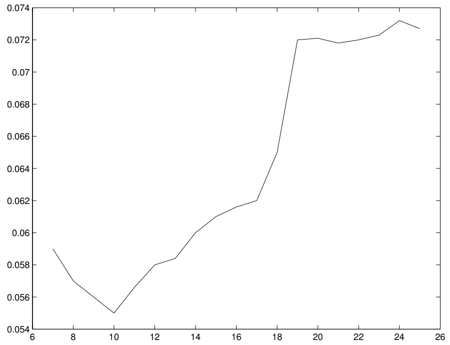

Figure 6: Error rates when the number of iteration changes

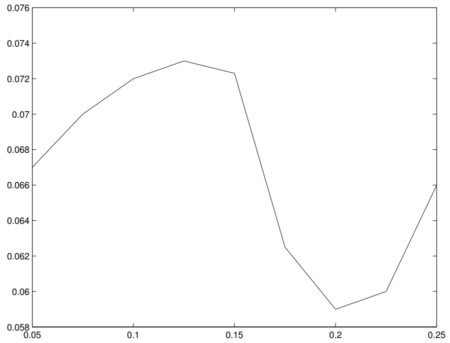

Figure 7: Error rates when the lambda changes

ument. The anisotropic diffusion technique is effective in enhancing the (sub)topic cohesion for text streams and documents.

In the anisotropic diffusion algorithm, there are three parameters, the number of iteration, the λ, and the conduction coefficient, that affect the quality of distance matrix segmentation improvement in various aspects. We study their impacts on segmentation performance by varying those parameters over certain ranges. Figure 6 shows the error rate changes when the number of iteration varies. Figure 7 shows the error rate changes when the λ varies. Our experiments show that the segmentation performance is in certain degree sensitive to the choices of those parameters. However the dependence on the conduction coefficient is very small and therefore we do not presented here. In all of our experiments we have used the set of parameter values in Table 5.

We have also computed the error rates for Brown corpus data set based on the Pevzner and Hearst's evaluation metric. They are listed in Table 6 for future research comparison. The average error rate of $Mars$ data set is 12% when the number of segments is known, and 16.9% when the number of segments is unknown.

# of iterations	conduction coefficient	lambda
10	20	0.2

Table 5: Parameters setting for anisotropic diffusion

	3-11	3-5	6-8	9-11
Our method'	7.0%	7.5%	5.6%	5.2%

Table 6: Error rate for the first data set calculated with second evaluation matric when the numbers of segments are known

4.4 Computational Complexity

The time cost of our method mainly comes from pairwise sentence distance calculation, anisotropic diffusion process, and the dynamic programming. The pairwise sentence distance calculation takes $O(n^2)$, and n is the number of sentences. The anisotropic diffusion process is $O(mn^2)$, where m is the number of iteration in the anisotropic diffusion process. The dynamic programming takes $O(kn^2)$, where k is the number of segments. In practice, m is a constant between five and ten in our experiments. The whole process takes about 10.5 seconds for 50 80-sentence documents topic segmentation on a 900MHZ, 1GByte memory SUN workstation with program implemented with Perl language.

5. CONCLUSION AND FUTURE WORK

We found that dynamic programming technique performs relatively well by itself. However, directly applying it to the distance matrix did not generate the best segmentation results. The reason is that complexity in lexical information blurs the topical cohesion in sentence groups. Thus, there are a significant amount of noise in the distance matrix. In order to reduce the noise as well as sharpen boundaries, we employ the anisotropic diffusion technique, which is the essential step in our method to reduce noise inside each sentence topical groups while keeping the topical cohesion of them well. In addition to generic stop words, we remove the document-dependent words as well. The experiment results demonstrate that the document-dependent words are very important to reduce the content correlation among sentences in long documents.

Some researchers developed text segmentation methods targeting at data sets from TDT corpus [19]. These data sets are text streams with only two or three sentences in each segment. Experimental results above indicate that our method needs to be improved in order to effectively segment this data set. We will investigate on this kind of data and improve our method for them. The document-dependent stop words will be through studies for potential impacts on different information retrieval tasks. The performance of our method will need to be optimized. We hope to speed up the current method with the current hardware configuration.

6. ACKNOWLEDGMENT

The research was supported in part by NSF grant CCR-9901986. The authors also want to thank Yixin Chen, Ya Zhang, and the referees for their constructive comments and suggestions.

7. REFERENCES

[1] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang. Topic Dectection and Tracking pilot study final report. In Proceedings of the DARPA broadcast news transcription and understanding workshop, pp.194-218, 1998.

[2] D. Beeferman, A. Berger, J. Lafferty. Statistical model for text segmentation. Machine Learning 34(1-3): 177-210 1999.

[3] B. K. Bogurae and M. S. Neff. Discourse Segmentation in Aid of Document Summarization. In Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000.

[4] D. M. Blei and P. J. Moreno. Topic segmentation with an aspect hidden markov model. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp.343-348, 2001.

[5] T. Brants, F. Chen, and I. Tsochantarides. Topic-based document segmentation with probabilistic latent semantic analysis. In Proceedings of the 11th International Conference on Information and Knowledge Management, pp.211-218, 2002.

[6] J. P. Callan. Passage-Level Evidence in Document Retrieval. In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.302-310, 1994.

[7] F. Y. Y. Choi. Advances in domain independent linear text segmentation. In Proceedings of NAACL, pp.26-33, 2000.

[8] M. Hajime, H. Takeo, and O. Manabu. Text segmentation with multiple surface linguistic cues. In Proceedings. of COLING-ACL, pp.881-885, 1998.

[9] J. A. Hartigan. Clustering Algorithms. *John Wiley & Sons*, New York, 1975.

[10] M. A. Hearst. Mult-paragraph segmentation of expository text. In Proceedings of the ACL'94, pp.9-16, 1994.

[11] M. A. Hearst. TextTiling:Segmenting text into multi-paragraph subtopic passages. Computational Linguistics, 23(1):33-64 1997.

[12] O. Heinonen. Optimal multi-paragraph text segmentation by dynamic programming. In Proceedings. of COLING-ACL'98, pp.1484-1486, 1998.

[13] T. K. Ho. Stop Word Location and Identification for Adaptive Text Recognition. International Journal of Document Analysis and Recognition, 3, 1, August 2000.

[14] X. Ji and H. Zha. Extracting Shared Topics of Multiple Documents. In Proceedings of the 7th Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2003.

[15] M. Y. Kan, J. L. Klavans, and K. R. McKeown. Linear segmentation and segment significance. In Proc. of the 6th International Workshop of Very Large Corpora, pp.197-205, 1998.

[16] H. Kozima. Text Segmentation based on similarity between words. In Proc. of the 31st Annual Meeting of the Association for Computational Linguistics, pp.286-288, 1993.

[17] I. Mani. Automatic Summarization. John Benjamins Pub Co., 2001.

[18] P. Perona and J. Malik. Scale-Space and Edge Detection Using Anisotropic Diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence Vol.12, No. 7, pp.629-639, 1990.

[19] J. M. Ponte and W. B. Croft. Text Segmentation by Topic. In Proc. of European Conference on Research and Advanced Technology for Digital Libraries, pp.120-129, 1997.

[20] M. Porter. The Porter Stemming Algorithm. www.tartarus.org/ martin/PorterStemmer

[21] L. Pevzner and M. Hearst. A Critique and Improvement of an Evaluation Metric for Text Segmentation. Computational Linguistics, 28 (1), pp.19-36, 2002.

[22] J. C. Reynar. Statistical models for topic segmentation. In Proc. of the 37th Annual Meeting of ACL, pp.357-364, 1999.

[23] G. Salton, J. Allan, and C. Buckley. Approaches to passage retrieval in full text information systems. In Proc. of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.49-58, 1993.

[24] G. Salton, A. Singhal, C. Buckley, and M. Mitra. Automatic Text Decomposition Using Text Segments and Text Themes. In Proc. of the Hypertext '96 Conference, pp.53-65, 1996.

[25] M. Utiyama and H. Isahara. A statitical model for domain-independent text segmentation. In Proc. of the 39th Annual Meeting of the Association for Computational Linguistics, pp.491-498, 1999.

[26] C. Wayne. Multilingual Topic Detection and Tracking: Successful Research Enabled by Corpora and Evaluation. In Proc. of Language Resources and Evaluation Conference, pages 1487-1494, 2000

[27] Y. Yamron. Segmentation of expository texts by hierarchical agglomerative clustering. In Proc. of Recent Advances in Natural Language Proceessings, 1997.

[28] H. Zha. Generic Summarization and Keyphrase Extraction Using Mutual Reinforcement Principle and Sentence Clustering. In Proc. of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.113-120, 2002.

[29] H. Zha and X. Ji. Correlating Multilingual Documents via Bipartite Graph Modeling. In Proc. of the 25th Annual International Conference on Research and Development in Information Retrieval, pp.443-444, 2002.