Using Multiple Discriminant Analysis Approach for Linear Text Segmentation

Zhu Jingbo[1], Ye Na[1], Chang Xinzhi[1], Chen Wenliang[1], and Benjamin K Tsou[2]

[1] Natural Language Processing Laboratory,
Institute of Computer Software and Theory, Northeastern University, Shenyang, P.R. China
{zhujingbo, chenwl}@mail.neu.edu.cn
{yena, changxz}@ics.neu.edu.cn
[2] Language Information Sciences Research Centre,
City University of Hong Kong, HK
rlbtsou@cityu.edu.hk

Abstract. Research on linear text segmentation has been an on-going focus in NLP for the last decade, and it has great potential for a wide range of applications such as document summarization, information retrieval and text understanding. However, for linear text segmentation, there are two critical problems involving automatic boundary detection and automatic determination of the number of segments in a document. In this paper, we propose a new domain-independent statistical model for linear text segmentation. In our model, Multiple Discriminant Analysis (MDA) criterion function is used to achieve global optimization in finding the best segmentation by means of the largest word similarity within a segment and the smallest word similarity between segments. To alleviate the high computational complexity problem introduced by the model, genetic algorithms (GAs) are used. Comparative experimental results show that our method based on MDA criterion functions has achieved higher P_k measure (Beeferman) than that of the baseline system using TextTiling algorithm.

1 Introduction

Typically a document is concerned with more than one subject, and most texts consist of long sequences of paragraphs with very little structural demarcation. The goal of linear text segmentation is to divide a document into topically-coherent sections, each corresponding to a relevant subject. Linear text segmentation has been applied in document summarization, information retrieval, and text understanding. For example, in recent years, passage-retrieval techniques based on linear text segmentation, are becoming increasingly popular in information retrieval as relevant text passages often provide better answers than complete document texts in response to user queries[1].

In recent years, many techniques have been applied to linear text segmentation. Some have used linguistic information[2,3,4,5,6,9] such as cue phrases, punctuation marks, prosodic features, reference, and new words occurrence. Others have used statistical methods[7,8,10,11,12,13,14,15] such as those based on word co-occurrence, lexical cohesion relations, semantic network, similarity between adjacent parts of texts, similarity between all parts of a text, dynamic programming algorithm, and HMM model.

R. Dale et al. (Eds.): IJCNLP 2005, LNAI 3651, pp. 292 – 301, 2005.

In linear text segmentation study, there are two critical problems involving automatic boundary detection and automatic determination of the number of segments in a document. Some efforts have focused on using similarity between adjacent parts of a text to solve topic boundary detection. In fact, the similarity threshold is very hard to set, and it is very difficult to identify exactly topic boundaries only according to similarity between adjacent parts of a text. Other works have focused on the similarity between all parts of a text. Reynar[7] and Choi[13] used *dotplots* technique to perform linear text segmentation which can be seen as a form of approximate and local optimization. Yaari[16] has used agglomerative clustering to perform hierarchical segmentation. Others[10,17,18,19] used dynamic programming to perform exact and global optimization in which some prior parameters are needed. These parameters can be obtained via uninformative prior probabilities[18], or estimated from training data[19].

In this paper, we propose a new statistical model for linear text segmentation, which uses Multiple Discriminant Analysis (MDA) method to define a global criterion function for document segmentation. Our method focuses on within-segment word similarity and between-segment word similarity. This process can achieve global optimization in addressing the two aforementioned problems of linear text segmentation. Our method is domain-independent and does not use any training data.

In section 2, we introduce Multiple Discriminant Analysis (MDA) criterion functions in detail. In section 3, our statistical model of linear text segmentation is proposed. A new MDA criterion function revised by adding penalty factor is further discussed in section 4. Comparative experimental results are given in Section 5. At last, we address conclusions and future work in section 6.

2 MDA Criterion Function

In statistical pattern classification, MDA approach is commonly used to find effective linear transformations[20,21]. The MDA approach seeks a projection that best separates the data in a least-squares sense. As shown in Figure 1, using MDA method we could get the greatest separation over data space when average within-class distance is the smallest, and average between-class distance is the largest.

Similarly, if we consider a document as data space, and a segment as a class, the basic idea of our approach for linear text segmentation is to find best segmentation of a document(greatest separation over data space) by focusing on within-segment word similarity and between-segment word similarity. It is clear that the smaller the average within-class distance or the average between-class distance, the larger the within-segment word similarity or the between-segment word similarity, and vice versa. In other words, we want to find the best segmentation of a document in which within-segment word similarity is the largest, and between-segment word similarity is the smallest. To achieve this goal, we introduce a criterion function to evaluate the segmentation of a document and assign a score to it. In this paper, we adopt the MDA approach to define a global criterion function of document segmentation, and called as MDA criterion function, which is described below.

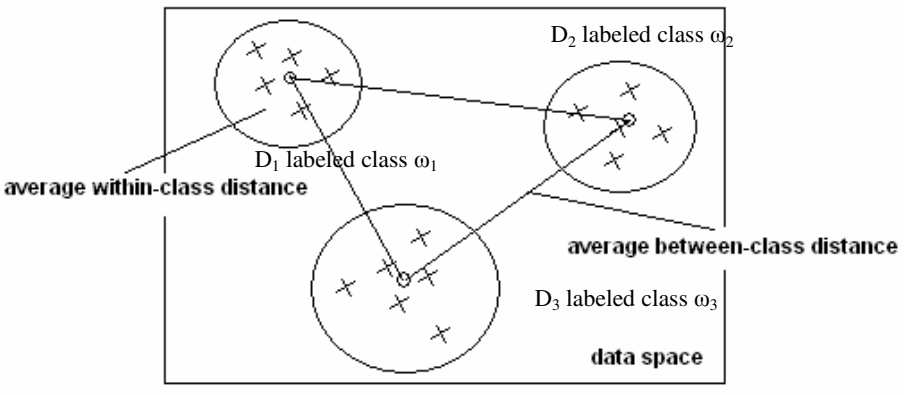

Fig. 1. When average within-class distance is the smallest, and average between-class distance is the largest, the greatest separation over data space is shown

Let $W=w_1w_2\ldots w_t$ be a text consisting of *t* words, and let $S=s_1s_2\ldots s_c$ be a segmentation of W consisting of *c* segments. We define W as data space, S as segmentation distribution over data space W. Because the lengths of paragraphs or sentences can be highly irregular, unbalanced comparisons can result in text segmentation process. Thus we adopt the *block* method that is used in the TextTiling algorithm[2,3], but we replace lexical word with block. In our model, we group *blocksize* words into a block which can be represented by a d-dimensional vector. In practice, we find that the value of *blocksize*=100 works well for many Chinese documents. Then $W=w_1w_2\ldots w_t$ can be redefined as $B=b_1b_2\ldots b_k$. As illustrated in Figure 1, a cross point can be defined as a d-dimensional block vector.

In this paper, we introduce MDA criterion function J_d in the following form[20]

$$J_d(s)=\frac{tr(S_B)}{tr(S_W)} \tag{1}$$

Where *tr(A)* is the trace of matrix A. S_W and S_B are within-segment scatter matrix and between-segment scatter matrix, respectively. S_W is defined by

$$S_W=\sum_{i=1}^{c}P_i\frac{1}{n_i}\sum_{b\in s_i}(b-m_i)(b-m_i)^t \tag{2}$$

Where b stands for blocks belonging to segment s_i, P_i is the a priori probability of segment s_i, and is defined to be the ratio of blocks in segment s_i divided by the total number of blocks of the document, n_i is the number of blocks in the segment s_i, m_i is the d-dimensional block mean of the segment s_i given by

$$m_i=\frac{1}{n_i}\sum_{b\in s_i}b. \tag{3}$$

Suppose that a total mean vector m is defined by

$$m = \frac{1}{n}\sum_{B} b = \frac{1}{n}\sum_{i=1}^{c} n_i m_i \tag{4}$$

In equation (1), between-segment scatter matrix S_B is defined by

$$S_B = \sum_{i=1}^{c} P_i (m_i - m)(m_i - m)^t \tag{5}$$

3 Statistical Model for Linear Text Segmentation

Using the same definitions of text W, segmentation S and blocks B in section 2, we first discuss the statistical model for linear text segmentation. The key of statistical model for text segmentation is to find the segmentation with maximum-probability. This can be turned into another task of finding segmentation with highest J_d score equally. The most likely segmentation is given by

$$\hat{S} = \arg\max_{S} \mathrm{P}(S \mid W) \overset{def}{=} \arg\max_{S} J_d(W, S) \tag{6}$$

As mentioned above, because paragraph or sentence length can be highly irregular, it leads to unbalanced comparisons in text segmentation process. So $W = w_1 w_2 \ldots w_n$ could be redefined as $B = b_1 b_2 \ldots b_k$, and the most likely segmentation is given by

$$\hat{S} = \arg\max_{S} \mathrm{P}(S \mid B) \overset{def}{=} \arg\max_{S} J_d(B, S) \tag{7}$$

The computational complexity for achieving the above solution is $O(2^k)$, where k is the number of blocks in a document. To alleviate the high computational complexity problem, we adopt the genetic algorithms (GAs)[22]. GAs provides a learning method motivated by an analogy to biological evolution. Rather than searching from general-to-specific hypotheses, or from simple-to-complex, GAs generate successor hypotheses by repeatedly mutating and recombining parts of the best currently known hypotheses. GAs have most commonly been applied to optimization problems outside machine learning, and are especially suited to tasks in which hypotheses are complex.

By adopting this methodology, we derive the following text segmentation algorithm, as illustrated in Figure 2. In this paper, we focus our study on paragraph-level linear text segmentation, in which the potential boundary mark between segments can be placed only between adjacent paragraphs.

Given a text W and blocks B, K_{max} is the total number of paragraphs in the text.
Initialization: $S_{best} = \{\}$, $J_d(B,S_{best})=0.0$
Segmentation:
For k = 2 **to** K_{max}
Begin
1) Use genetic algorithms and equation (7) to find the best segmentation S of k segments.
2) **If** $J_d(B,S_{best}) < J_d(B,S)$ **Then**
Begin
$S_{best} = S$ and $J_d(B,S_{best}) = J_d(B,S)$.
Endif
Endfor
Output the best segmentation S_{best}.

Fig. 2. MDA-based text segmentation algorithm

4 Penalty Factor

In the text segmentation process, adjacent boundary adjustment should be considered in cases when there are some very close adjacent but incorrect segment boundaries. In experiments we find that in these cases some single-sentence paragraphs are wrongly recognized as isolated segments. To solve the problem, we propose a penalty factor (PF) to prevent assignment of very short segment boundaries (such as a single-sentence segment) by adjusting very close adjacent boundaries, and therefore improve the performance of linear text segmentation system.

Suppose that we get a segmentation $S=s_1s_2\ldots s_c$ of the input document, let L be the length of the document, L_i be the length of the segment s_i. We know $L=L_1+L_2+\ldots+L_c$. We define penalty factor as

$$PF = \prod_{i=1}^{c} \frac{L_i}{L} \tag{8}$$

As can be seen, short-length segments would result in smaller penalty factor. We use penalty factor to revise the J_d scores of segmentations. To incorporate the penalty factor PF, our MDA criterion function J_d can be rewritten as

$$J_{d-PF}(x) = PF \times J_d(x) = \prod_{i=1}^{c} \frac{L_i}{L} \times \frac{tr(S_B)}{tr(S_W)} \tag{9}$$

In the following experiments, we will evaluate effectiveness of using the two MDA criterion functions J_d and $J_{d\text{-}PF}$ for linear text segmentation.

5 Experimental Results

5.1 Evaluation Methods

Precision and *recall* statistics are conventional means of evaluating the performance of classification algorithms. For the segmentation task, *recall* measures the fraction of actual boundaries that an automatic segmenter correctly identifies, and *precision* measures the fraction of boundaries identified by an automatic segmenter that are actual boundaries. The shortcoming is that every inaccurately estimated segment boundary is penalized equally whether it is near or far from a true segment boundary.

To overcome the shortcoming of *precision* and *recall*, we use a measure called P_k, proposed by Beeferman *et al.*[8]. P_k method measures the proportion of sentences which are wrongly predicted to belong in the same segment or sentences which are wrongly predicted to belong in different segments. More formally, given two segmentations *ref*(true segmentation) and *hyp*(hypothetical segmentation) for a document of n sentences, P_k is formally defined by

$$P_k(ref, hyp) = \sum_{1 \le i \le j \le n} D_\mu(i, j)(\delta_{ref}(i, j) \overline{\oplus} \delta_{hyp}(i, j)) \tag{10}$$

Where $\delta_{ref}(i,j)$ is an indicator function whose value is 1 if sentences i and j belong in the same segment in the true segmentation, and 0 otherwise. Similarly, $\delta_{hyp}(i,j)$ is an indicator function which evaluates to 1 if sentences i and j belong in the same segment in the hypothetical segmentation, and 0 otherwise. The operator between $\delta_{ref}(i,j)$ and $\delta_{hyp}(i,j)$ in the above formula is the *XNOR* function on its two operands. The function D_μ is a distance probability distribution over the set of possible distances between sentences chosen randomly from the document, and will in general depend on certain parameters μ such as the average spacing between sentences. In equation (10), $D_\mu$was defined as an exponential distribution with mean $1/\mu$, a parameter that we fix at the approximate mean document length for the domain[8].

$$D_\mu(i, j) = \gamma_\mu e^{-\mu|i-j|} \tag{11}$$

Where $\Upsilon_\mu$is a normalization chosen so that D_μ is a probability distribution over the range of distance it can accept. From the above formulation, we could find one weakness of the metric: there is no principled way of specifying the distance distribution D_μ. In the following experiments, we use P_k as performance measure, where the mean segment length in the test data was $1/\mu$=11 sentences.

5.2 Quantitative Results

We mainly focus our work on paragraph-level linear text segmentation techniques. The Hearst's TextTiling algorithm[2,3] is a simple and domain-independent technique for linear text segmentation, which segments at the paragraph level. Topic boundaries are determined by changes in the sequence of similarity scores. This algorithm uses a simple cutoff function to determine automatically the number of boundaries.

In our experiments, we use the TextTiling algorithm to provide the baseline system, and use the P_k measure to evaluate and compare the performance of the TextTiling and our method. Our data set - NEU_TS, is collected manually, and it consists of 100 Chinese documents, all from *2004-2005 Chinese People's Daily newspaper*. The number of segments per document varies from five to eight. The average number of paragraphs per document is 25.8 paragraphs. To build the ground truth for NEU_TS data set, five trained graduate students in our laboratory who are working on the analysis of Chinese document are asked to provide judgment on the segmentation of every Chinese document. We first use the toolkit CipSegSDK[23] for document preprocessing, including word segmentation, but with the removal of stopwords from all documents.

1) Experiment 1

In the first experiment, we assume the number of segments of an input document is known in advance. We use the NEU_TS data set and the P_k measure to evaluate and compare the performance of TextTiling and our method. The purpose of this experiment is to compare the performance of boundary detection techniques of TextTiling algorithm and our model using MDA criterion functions.

Table 1. P_k value with known number of document segments

Measure	TextTiling algorithm	MDA method using J_d	MDA method using $J_{d\text{-}PF}$
P_k value	0.825	0.869	0.905

In the TextTiling algorithm, topic boundaries are determined by changes in the sequence of similarity scores. The boundaries are determined by locating the lowermost portions of valleys in the resulting plot. Therefore, it is not a global evaluation method. However, in our model, MDA criterion function provides a global evaluation method to text segmentation; it selects the best segmentation with the largest within-segment word similarity and the smallest between-segment word similarity. Results shown in Table 1 indicated that our boundary detection techniques based on two MDA criterion functions perform better than the TextTiling algorithm, and MDA criterion function $J_{d\text{-}PF}$ works the best.

Table 2. P_k value with unknown number of document segments

Measure	TextTiling algorithm	MDA method using J_d	MDA method using $J_{d\text{-}PF}$
P_k value	0.808	0.831	0.87

2) Experiment 2

In this experiment, we assume the number of segments of a document is unknown in advance. In other words, Texttiling algorithm and our model should determine the number of segments of a document automatically. Similar to Experiment 1, the same

data set is used and the P_k measure is calculated for both TextTiling and our method using MDA criterion functions J_d and $J_{d\text{-}PF}$. The comparative results are shown in Table 2.

As mentioned above, how to determine the number of segments to be assigned to a document is a difficult problem. Texttiling algorithm uses a simple cutoff function method to determine the number of segments and it is sensitive to the patterns of similarity scores[2,3]. The cutoff function is defined as a function of the average and standard deviations of the depth scores for the text under analysis. A boundary is drawn only if the depth score exceeds the cutoff value. We think that the simple cutoff function method is hard to achieve global optimization when solving these two key problems of linear text segmentation process. In our model, two MDA criterion functions J_d and $J_{d\text{-}PF}$ are used to determine the number of segments and boundary detection by maximizing J_d score of segmentations. Once the maximum-score segmentation is found, the number of segments of the document is produced automatically. Experimental results show that our MDA criterion functions are superior to the TextTiling's cutoff function in terms of automatic determination of the number of segments. It is also shown that the MDA criterion function $J_{d\text{-}PF}$ revised with Penalty Factor works better than J_d. In implementation, we have adopted genetic algorithms (GAs) to alleviate the computational complexity of MDA, and have obtained good results.

6 Conclusions and Future Work

In this paper, we studied and proposed a new domain-independent statistical model for linear text segmentation in which multiple discriminant analysis(MDA) approach is used as global criterion function for document segmentation. We attempted to achieve global optimization in solving the two fundamental problems of text segmentation involving automatic boundary detection and automatic determination of number of segments of a document, by focusing on within-segment word similarity and between-segment word similarity. We also applied genetic algorithms(GAs) to reduce the high computational complexity of MDA based method. Experimental results show that our method based on MDA criterion functions outperforms the TextTiling algorithm.

The solution to the high computational complexity problem will continue to be studied by using other effective optimization algorithm or near optimal solutions. In the next stage we plan to combine MDA criterion functions with other algorithms such as clustering to improve the performance of our text segmentation system, and apply the text segmentation technique to other text processing task, such as information retrieval and document summarization.

Acknowledgements

We thank Keh-Yih Su and Matthew Ma for discussions related to this work. This research was supported in part by the National Natural Science Foundation of China & Microsoft Asia Research Centre(No. 60203019), the Key Project of Chinese Ministry of Education(No. 104065), and the National Natural Science Foundation of China(No. 60473140).

References

1. Gerard Salton, Amit Singhal, Chris Buckley, and Mandar Mitra.: Automatic text decomposition using text segments and text themes. In proceedings of the seventh ACM conference on Hypertext, Bethesda, Maryland, United States (1996) 53-65
2. Hearst, M.A.: Multi-paragraph segmentation of expository text. In proceedings of the 32th Annual Meeting of the Association for Computational Linguistics, Las Cruces, New Mexico (1994) 9-16
3. Hearst, M.A.: TextTiling: segmenting text into multi-paragraph subtopic passages. Computational Linguistics, Vol.23, No.1 (1997) 33-64
4. Youmans, G.: A new tool for discourse analysis: The vocabulary management profile. Language, Vol.67, No.4 (1991) 763-789
5. Morris, J. and Hirst, G.: Lexical cohesion computed by thesauri relations as an indicator of the structure of text. Computational Linguistics, Vol.17, No.1 (1991) 21-42
6. Kozima, H.: Text segmentation based on similarity between words. In proceedings of the 31th Annual Meeting of the Association for Computational Linguistics, Student Session (1993) 286-288
7. Reynar, J.C.: An automatic method of finding topic boundaries. In proceedings of the 32 nd Annual Meeting of the Association for Computational Linguistics, Student Session, Las Cruces, New Mexico (1994) 331-333
8. Beeferman, D., Berger, A., and Lafferty, J.: Text segmentation using exponential models. In proceedings of the Second Conference on Empirical Methods in Natural Language Processing, pages, Providence, Rhode Island (1997) 35-46
9. Passoneau, R. and Litman, D.J.: Intention-based segmentation: Human reliability and correlation with linguistic cues. In proceedings of the 31st Meeting of the Association for Computational Linguistics (1993) 148-155
10. Jay M. Ponte and Bruce W. Croft.: Text segmentation by topic. In proceeding of the first European conference on research and advanced technology for digital libraries. U.Mass. Computer Science Technical Report TR97-18 (1997)
11. Reynar, J.C.: Statistical models for topic segmentation. In proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (1999) 357-364
12. Hirschberg, J. and Grosz, B.: Intentional features of local and global discourse. In proceedings of the Workshop on Spoken Language Systems (1992) 441-446
13. Freddy Y. Y. Choi.: Advances in domain independent linear text segmentation. In Proc. of NAACL-2000 (2000)
14. Choi, F.Y.Y., Wiemer-Hastings, P. & Moore, J.: Latent semantic analysis for text segmentation. In proceedings of the 6th Conference on Empirical Methods in Natural Language Processing (2001) 109-117.
15. Blei, D.M. and Moreno, P.J.: Topic segmentation with an aspect hidden Markov model. Tech. Rep. CRL 2001-07, COMPAQ Cambridge Research Lab (2001)
16. Yaari, Y.: Segmentation of expository texts by hierarchical agglomerative clustering. In proceedings of the conference on recent advances in natural language processing (1997) 59-65
17. Heinonen, O.: Optimal multi-paragraph text segmentation by dynamic programming. In proceedings of 17th international conference on computational linguistics (1998) 1484-1486.
18. Utiyama, M., and Isahara, H.: A statistical model for domain-independent text segmentation. In proceedings of the 9th conference of the European chapter of the association for computational linguistics (2001) 491-498

19. A Kehagias, P Fragkou, V Petridis.: Linear Text Segmentation using a Dynamic Programming Algorithm. In proceedings of 10th Conference of European chapter of the association for computational linguistics (2003)
20. R. Duda, P. Hart, and D. Stork.: Pattern Classification. Second Edition, John Wiley & Sons (2001)
21. Julius T.Tol and Rafael C. Gonzaiez.: Pattern recognition principles. Addison-Wesley Publishing Company (1974)
22. Tom M.Mitchell.: Machine Learning. McGraw-Hill (1997)
23. Yao Tianshun, Zhu Jingbo, Zhang li, and Yang Ying.: Natural language processing-research on making computers understand human languages. Tsinghua university press (2002)