A Preliminary Study of Lexical Density for the Development of XML-based Discourse Structure Tagger

Lawrence Y. L. Cheung[*], Tom B. Y. Lai[*], Benjamin K. Tsou[*],
Francis C. Y. Chik[*], Robert W. P. Luk[§], Oi Yee Kwong[*]

[*]Language Information Sciences Research Centre
City University of Hong Kong
Kowloon Tong, Hong Kong

{rlylc, cttomlai, rlbtsou, rlfchik,
rlolivia}@cityu.edu.hk

[§]Department of Computing
Hong Kong Polytechnic University
Hung Hom, Hong Kong

csrluk@comp.polyu.edu.hk

Abstract

The identification of discourse segments is crucial to many NLP tasks, particularly, summarization. This paper discusses the development of an XML-based Discourse Structure Tagger based on the distribution of legal terminology in Chinese judgment texts. A computational approach is proposed to automatically identify the major segment breaks. The method involves the use of a difference metric for detecting significant changes in lexical density and the use of *K*-means clustering technique for selecting most probable inter-segmental break locations.

Keywords: Discourse Structure, Information Extraction, XML

1 Introduction

The growing volume of electronic documentation has created a need for text retrieval and summarization. Mani and Maybury (1999) have identified three major approaches to text summarization, namely surface, entity, or discourse levels. Surface-level approaches make use of shallow features (salient terms, location, cue phrases, etc.) to capture textual information and compute which unit to be extracted. Entity-level approaches build an internal representation for connectivity in the text, e.g. vocabulary overlap, term co-occurrence, syntactic relations, etc. Discourse-level approaches model topic development and rhetorical structure of the text.

There are obvious advantages of discourse-based approach. Knowledge about the discourse structure provides clues to message type, and facilitates the utilization of relevant portions of the texts. However, many discourse-based frameworks have made minimal assumptions about inherent discourse structure of different text genres. Discourse structure is built on notions of cohesion and coherence such as Barzilay and Elhadad's (1997) lexical chain and Marcu's (2000) rhetorical structure parsing for intra-sentential relationship. While the approach has the advantage of generality, significant properties pertaining to specific text genres such as term distribution and discourse communicative goal may have been overlooked. As Sparck-Jones (1999) has pointed out, in the absence of an adequate theory of discourse structure for summarization, it is important to consider operational factors (e.g. text structure, genre, audience, etc.) for any particular application of text summarization.

Useful as discourse segmentation is, seldom do annotated documents capture discourse structure information. This paper studies techniques for automatic identification and tagging of discourse structure of judgment texts stored in XML format. XML has become an emerging markup language for describing the structure of information in general. It is also an ideal platform for text annotation and dynamic text access. While not a summarization study per se, the discourse structure analysis will facilitate the development of an *XML-based discourse structure tagger*, and eventually enrich the input for other NLP tools such as full-document summarizers.

The paper will first discuss the discourse structure of Chinese legal judgment texts. Based on the findings, we will then present a computational approach that makes use of the knowledge of macro-structure of judgment texts to compute segment breaks. In particular, we are

interested in the identification of the four major units in judgment texts—*opening*, *facts*, *reasoning* and *verdict*.

2 Background

2.1 Legal Information Retrieval

In the Common Law system, precedents constitute an essential basis in legal argument. Court judgments are cited as an example or analogy to justify decisions made for cases alike. The barrister or the litigation clerk has to search for relevant cases to support his argument. The availability of digitized legal documents makes it possible to retrieve these cases by computational means. Ashley (1990) describes the AI program *Hypo*, which uses legal case knowledge frames, case attribute indices and reasoning logic to support search on "trade-secret" cases. However, the knowledge-rich design depends on extensive manual crafting of rules of a confined domain, and is highly domain-specific.

Text summarization and information extraction are promising alternative to facilitate search. They provide a more generalized and adaptable approach for fairly accurate case retrieval system. For example, Moven (2000) reports that summarization techniques have been utilized in SALOMON project for providing retrieval system for various domains of Belgian criminal cases.

2.2 Discourse Structure

Knowledge about the discourse structure of a text can increase processing precision by reducing search space, and facilitate the utilization of relevant portions of the texts by other NLP tasks such as text categorization, summarization, document retrieval, and so on.

Depending on communicative functions and genres, e.g. email, scientific report, and newspaper articles, texts exhibit distinctive discourse structures and patterns of message distribution. Indeed the writer is often guided by explicit or implicit convention. For example, a scientific paper normally develops in the following pattern: *introduction, methodology, experiments, results, discussion* and *conclusion*. Since the communicative goals of segments govern the distribution of content and presentation style, the identification of the discourse organization can significantly facilitate text processing. Paragraph breaks alone are not adequate to demarcate discourse segments. A discourse segment may be embedded in paragraphs together with other units, or may span across several paragraphs.

The exploitation of discourse structure in text summarization is not novel. Boguraev and Kennedy (1999) use parsing techniques to find out the cohesion relation of anaphoric expressions. Hearst (1997) computes discourse salience based on similarity between adjacent text blocks. Barzilay and Elhadad (1997) capitalize on the lexical chains to trace the cluster of textual units. Marcu (2000) applies rhetorical structure tree to model text coherence. These studies deal primarily with general texts, and assume little about the text genre properties. The approach proposed in Section 4 takes advantage of discourse features pertaining to the judgment genre to locate discourse segments. Since the application is tied to the judgment genre but not case domain, it is considered to be semi-domain-independent.

3 Characteristics of Judgment Discourse

3.1 Legal Judgements in Hong Kong

After the return of the sovereignty of Hong Kong to China in 1997, the common law system is still retained in Hong Kong. Chinese was recognised as one of the two official languages on par with English in early 90s. However, the majority of the judgments are still written in English. If necessary, the Chinese translation will also be provided. To promote and facilitate legal bilingualism, a project has been undertaken by City University of Hong Kong in collaboration with the Hong Kong Judiciary to design and implement (*E*lectronic *L*egal *Do*cumentation/Corpus *S*ystem) ELDoS, a bilingual judgment retrieval system (Kwong et al., 2001). Authentic Chinese judgments from the system are used for the present study.

3.2 Discourse Segments

Legal judgment is a written document about the decision of a court of law or a judge prepared by the judge. The length may vary from 1 to 30+ pages. Its language style is generally less formal and technical than that found in laws. Over the years, a conventional structure of legal judgment

has been developed. A typical judgment can be divided into four major functional segments.

1. ***Opening (OP)***, a brief summary of the parties involved, application request, nature of the case, etc.
2. ***Factual Elucidation (FE)***, a descriptive highlight of the alleged offences, factual evidence, etc.
3. ***Reasoning (RE)***, stating issues of the case, arguments and opinion of the court, and legal foundations, and statutory provisions applied by the court
4. ***Verdict (VD)***, the finding of a jury in a trial and an optional sentencing

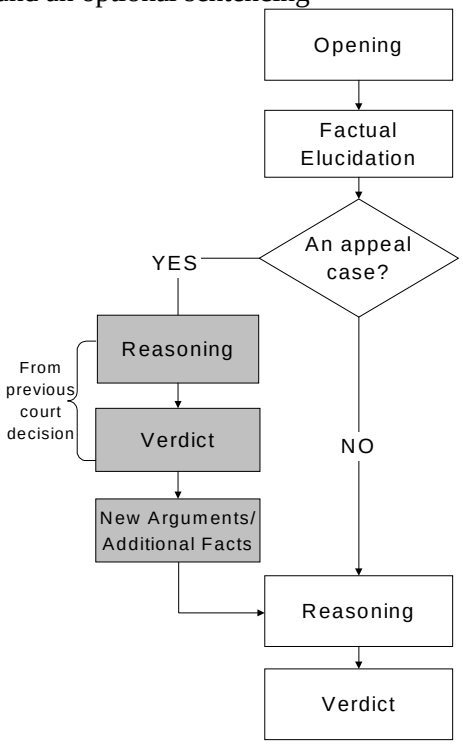

Figure 1. Segment Sequence in a Judgment

A judgment basically follows the sequence—*Opening*, *Factual Elucidation*, *Reasoning*, and *Verdict*. However, in appeal cases, the reasoning, verdict of lower level court(s), new arguments will follow *Factual Elucidataion* immediately. For simplicity, the units pertaining to appeal cases will not be dealt with here. The segment sequence in judgment is generalized in Figure 1.

3.3 Linguistic Features

The judgment segments manifest rather different linguistic cues.

Opening is a formal summary about the case at the beginning of a judgment. It is condensed and loaded with legal terminology.

Factual Elucidation is located early in a judgment. The language tends to be descriptive, less technical and less formal. Time, place and manner adverbials are commonly found. In English judgment, facts are usually reported using past tense.

Reasoning accounts for about half to two-thirds of the entire judgment. The segment contains more legal terminology and formal expressions. Linguistic cues for this segment include hypothetical and conditional statements. Citations of precedents (e.g. "Chim Hon Man v. HKSAR (1999) 2 HKCFAR 145") are frequently found in legal reasoning.

Verdict occurs at the end of a judgment text. As the function is very specific, the lexical items are very restricted. Words such as 駁回 (*dismiss*), 拒絕 (*refuse*), 裁定 (*hold*), etc. are indicative of occurrence of verdict.

The linguistic characteristics of the segments are summarized in Table 1.

Seg.	Location	Length	Vocab	Cues
OP	beginning	short	legal vocab	
FE	front to middle	medium	descriptive, common vocab	temporal expressions
RE	middle to end	medium to long	legal vocab	conditionals, hypotheticals connectives, citations
VD	at the end	short	restricted, legal vocab	

Table 1. Summary of segment characteristics

4 Identification of Segments

Despite the range of linguistic cues associated with different segments, this preliminary study will focus on the use of lexical distribution in the identification of judgment segments. The techniques are applied to Chinese judgment texts from Hong Kong Judiciary.

4.1 Lexical Density

The distribution of different lexical types is measured in terms of *Lexical Density (LD)*. Words and phrases in each sentence are compared with entries in two legal dictionaries, *GLDict* and *VLDict*. *GLDict* currently contains 400+ entries of "**General Legal Terminology**" (*GLT*); *VLDict* contains 20+ entries of "**Verdict Legal Terminology**" (*VLT*) that are typically found in verdict. *VLDict* is a subset of *GLDict*. The LD of *GLT* and *VLT* of sentence s_i are

defined below:

$LD_{GLT}(s_i) = (N_{GLT, Si} / N_{Si}) * \log_{10}(N_{Si})$

$LD_{VLT}(s_i) = (N_{VLT, Si} / N_{Si}) * \log_{10}(N_{Si})$

where

$N_{GLT, Si}$ = No. of *GLT* word tokens in s_i

$N_{VLT, Si}$ = No. of *VLT* word tokens in s_i.

N_{Si} = No. of word tokens in s_i

The multiplication of the term "$\log_{10}(N_S)$" is to produce a higher density value for longer sentences.

4.2 Identification Strategy

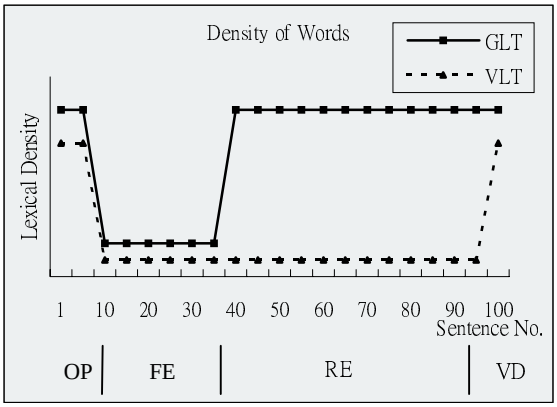

Figure 2. Idealized LD Graph for a Judgment

By observing the distribution of lexical density of sentences, we can determine the location of the segments. Figure 2 is an idealized model of lexical distribution in a judgment text based on observations in Section 3.3. Figure 3 shows the *GLT* and *VLT* lexical density of an authentic judgment text.

A strategy for the identification of discourse segments is presented as follows.

Step 1: Identification of *OP* & *FE*

Identify the *FE* segment which is distinguished from all other segments by its low GLT density (LD_{GLT}) value.

From Figure 2, to identify the *FE* segment, it is necessary only to locate the *OP-FE* break and *FE-RE* break. The two breaks are characterized by the drastic decrease and increase in the density of *GLT*. All sentences before *FE* segment are assigned to *OP* segment.

Step 2: Identification of *RE* & *VD*

Identify the *VD* segment which is located near the end and whose *VLT* density (LD_{VLT}) value is high.

From Figure 2, we need to locate the *RE-VD* break to identify *Verdict* segment. All sentences between *FE-RE* break and *RE-VD* break belong to *RE* segment.

In short, three breaks, namely *OP-FE*, *FE-RE* and *RE-VD*, have to be located.

4.3 Difference Metrics for Break Detection

According to Figure 2, *inter-segmental* breaks are characterized by the drastic change in the LD_{GLT}. An inter-segmental break may be declared when the LD_{GLT} difference between the current sentence and the following sentences exceeds a threshold *t*, i.e.

$$| LD_{GLT}(s_i) - LD_{GLT}(s_{i+1}) | > t$$

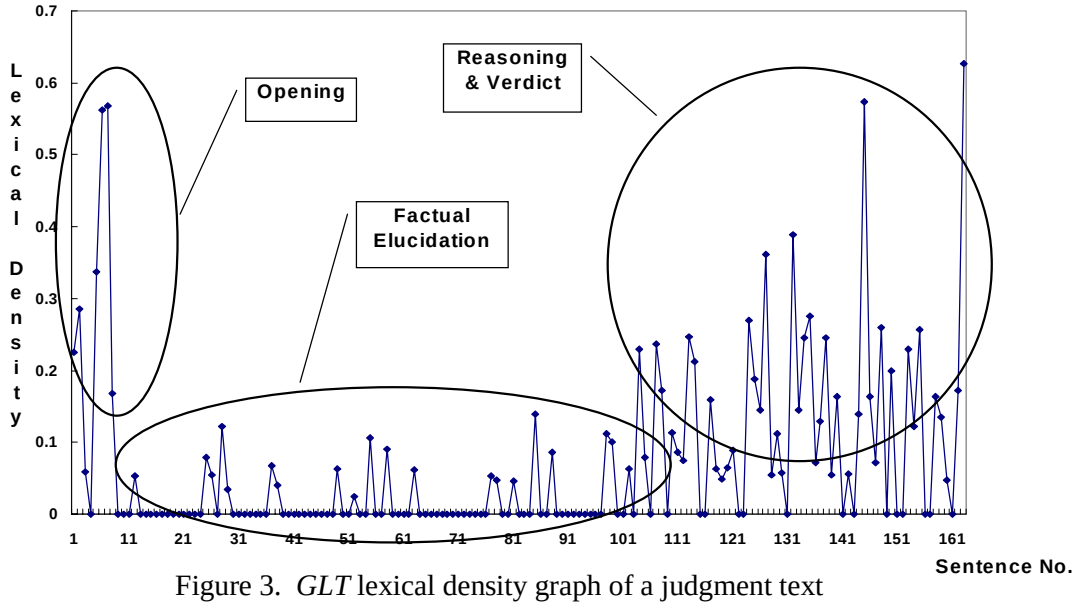

Figure 3. *GLT* lexical density graph of a judgment text

However, as Figure 2 has revealed, there is much *intra-segmental* fluctuation of lexical density. A potential problem with the above metric is its sensitivity to intra-segmental changes. A procedure has to be modified to distinguish between inter- and intra-segmental variations.

To avoid the interference of local fluctuation, the neighbourhood context of the sentence will be considered. The assumption is that the effect of local flunctuation can be averaged out when a wider context is considered. The density of a fixed number of sentences immediately preceding and following the current sentence is included in the difference metric forumla. We define the difference (LDDiff) as follows:

$$\mathrm{LDDiff}_{GLT}(s_i) = \sum_{u=0}^{k-1} \mathrm{LD}_{\mathrm{GLT}}(s_{i-u}) - \sum_{v=1}^{k} \mathrm{LD}_{\mathrm{GLT}}(s_{i+v})$$

where k is the window size

In our testing, a window size of 4 sentences before and after the sentence is found to be optimal. Maximums and minimums of $\mathrm{LDDiff}_{GLT}(x)$ graph imply significant changes of lexical density. The graph in Figure 4 is the LDDiff_{GLT} curve for the curve in Figure 3.

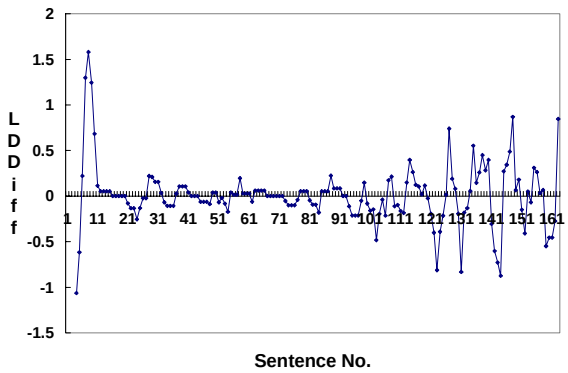

Figure 4. LDDiff_{GLT} graph for the curve in Fig 3.

4.4 *K*-means Clustering

K-means clustering (Anderberg, 1973; Cios et al., 1998) is an unsupervised non-heirarchial clustering commonly used for grouping together data points that are similar to each other. The method first assigns each data point to the nearest centroid (mean) of the *K* clusters. Based on the partition, the new centroids of each group are re-computed. The process is repeated until no data points change their cluster membership.

Inter-segmental breaks are likely to be local minimums and maximums in LDDiff graphs. Falling-edge (e.g. *OP-FE* break) in LD graph should correspond to a local maximum; and rising-edge (e.g. *FE-RE* break), a local minimum. To classify the points in LDDiff algorithmically, *K*-means clustering is applied to assign each point to one of the 3 clusters (i.e. K=3). $\mathrm{LDBOTTOM}_{GLT}$ and LDTOP_{GLT} clusters represent minimums and maximums with a large amplitude respectively. $\mathrm{LDMIDDLE}_{GLT}$ cluster groups together all other values mostly close to zero.

OP-FE break falls between s_i and s_{i+1} such that s_i is a member of LDTOP_{GLT} and the value of i is the smallest in the LDTOP_{GLT} cluster. *FE-RE* break falls between s_j and s_{j+1} such that s_j is a member of $\mathrm{LDBOTTOM}_{GLT}$ and j is the smallest in the $\mathrm{LDBOTTOM}_{GLT}$ cluster such that $i < j$. Similarly, the *RE-VD* break is located between s_i and s_{i+1} where s_i is a member of $\mathrm{LDBOTTOM}_{VLT}$ with the smallest i.

5 Experiment

5.1 Data Markup

Ten Chinese judgment texts are taken from the ELDoS corpus which encodes and alignes bilingual (Chinese and English) judgment texts in XML. Each text is marked up at four levels, i.e. document[1], paragraph, sentence, and word. XML tagging is done semi-automatically. As Chinese lacks explicit word boundary, automatic word segmentation and tagging is first applied, followed by human verification. The tagger then identifies and tags paragraphs and sentences by detecting sentence delimiter punctuation mark (e.g. "。") and newline character respectively.

5.2 XML Encoding

There are advantages in using XML to annotate text for language processing. **First**, developers have the flexibility to define the semantics and structure of XML documents to store information. Although this study focusses on finding 4 major segments in judgments, the flexibility permits future expansion and modification of segment tags. **Second**, XML content abstraction has been introduced by

[1] This is to distinguish between the Chinese and English version of the judgment. In this study, only the Chinese version is used.

World Wide Web Consortium [2] to facilitate reference to information found in an XML document. Many of these associated technologies and specifications are hierarchy-based, making them ideal to capture textual structure like paragraphs, sentences and words. Among others, XML comes with **Document Object Model** (DOM) (Le Hors et al., 2001) and **XML Path Language** (XPath) (Clark and DeRose, 1999). Both model XML documents as a tree data structure. DOM creates and hosts data in a logical hierarchical model of nodes based on the structures defined in XML documents. The contents are represented as a set of nodes, abstract information containers, in tree data structure. NLP tools can extract, insert, manipulate and navigate the data via the standardized interface. XPath is the convention to address parts of data in XML-based documents. This enables dynamic access to required data at different textual levels.

A judgment analyzer has been developed to extract words, sentences and paragraphs from the XML judgment files. It takes advantage of XPath and DOM Objects/ Interfaces implemented in MSXML Parser 3.0 Release[3] to traverse and process sentences in each paragraph. For example, the DOM method `selectNodes "/judgment/doc[@xml:lang='zh']/p[i]/s[j]"` to select the string for the *j*-th sentence in the *i*-th paragraph. Though English and Chinese text reside in the same document, XPath permits the program to skip English text by the attribute `[@xml:lang='zh']`.

The analyzer then stores the following attributes of each sentence for LD computation.

1. File no.
2. Paragraph no.
3. Sentence no.
4. Sentence string
5. No. of word tokens
6. No. of *GLT* word tokens
7. No. of *VLT* word tokens

5.3 Identification Procedures

Two procedures, `Ident_Facts` and `Ident_Verdict`, have been created for the identification of the three major breaks.

5.3.1 Some Definitions

Here are some some definitions used in the two procedures.

s_i = the *i*-th sentence in a judgment text

N_{Si} = no. of words in s_i

$LT \in \{GLT, VLT\}$

$N_{LT, Si}$ = no. of *LT* words in s_i

$$\text{LDDiff}_{LT}(s_i) = \sum_{u=0}^{k-1} \text{LD}_{LT}(i-u) - \sum_{v=1}^{k} \text{LD}_{LT}(s_{i+v})$$

$\text{LD}_{LT}(s_i) = (N_{LT, Si} / N_{Si}) * \log_{10}(N_{Si})$

where k (=4) is the window size

A sentence s_i is defined as an LDMAX_{LT} in the LDDiff_{LT} graph iff
(a) $\text{LDDiff}_{LT}(s_{i-1}) < \text{LDDiff}_{LT}(s_i)$ and $\text{LDDiff}_{LT}(s_i) \geq \text{LDDiff}_{LT}(s_{i+1})$, and
(b) $\text{LDDiff}_{LT}(s_i)$ is a member of the LDTOP_{LT} cluster (computed using *K*-means)

A sentence s_i is defined as an LDMIN_{LT} in the LDDiff_{LT} graph iff
(a) $\text{LDDiff}_{LT}(s_{i-1}) \geq \text{LDDiff}_{LT}(s_i)$ and $\text{LDDiff}_{LT}(s_i) < \text{LDDiff}_{LT}(s_{i+1})$, and
(b) $\text{LDDiff}_{LT}(s_i)$ is a member of the LDBOTTOM_{LT} cluster (computed using *K*-means)

5.3.2 `Ident_Facts` Procedure

Procedure

Procedure `Ident_Facts`

```
// initialization
 LDDiff_GLT(s_i) = 0 for 1 ≤ i ≤ 3
 Compute LDDiff_GLT(s_i) for i ≥ 4.

// find beginning of facts segment
 (Scan s_i in ascending order of i)
 1. Find the first s_i such that s_i is LDMAX_GLT.
 2. If there exists a paragraph break between s_i
    and s_{i+1} then
       i_b = i+1
    Else
       Find the biggest m such that
          a. there is a paragraph break
           between s_{m-1} and s_m,
          b. 0 ≤ i-m ≤ 2.
       If such s_m is found then
          i_b = m
       else
          i_b = i
       End if
    End if
 // i_b-th sentence is the beginning of FE
```

[2] http://www.w3.org

[3] http://msdn.microsoft.com/xml

// find end of *FE* segment

3. Find the first s_j such that s_j is $LDMIN_{GLT}$, where $j > i$.
4. If there is a paragraph break between s_{j-1} and s_j then
   $i_e = j-1$
   Else
   Find the smallest n such that
   a. there is a paragraph break between s_n and s_{n+1},
   b. $n \geq j$ and $0 \leq j - i \leq 2$.
   If such s_n is found then
   $i_e = n$
   else
   $i_e = j$
   End if
   End if

// i_e-*th* sentence is the end of *FE*
End Procedure

5.3.3 `Ident_Verdict` Procedure

Procedure
Procedure Ident_Verdict

// initialization
$LDDiff_{VLT}(s_i) = 0$ for $1 \leq i \leq 3$
Compute $LDDiff_{VLT}(s_i)$ for $i \leq 4$.

// find beginning of *VD* segment
(Scan s_i in descending order of i)

1. Find the first s_j such that s_j is $LDMIN_{VLT}$.
2. If there exists a paragraph break between s_j and s_{j+1} then
   $i_b = j+1$
   Else
   Find the smallest n such that
   a. there is a paragraph break between s_n and s_{n+1},
   b. $0 \leq n - j \leq 2$
   If such s_n is found then
   $i_b = n + 1$
   else
   $i_b = j$
   End if
   End if

// i_b-*th* sentence is the end of *VD*
End Procedure

Since inter-segmental breaks normally coincide with a paragraph break, the above procedures check if a paragraph break is close to (no more than 2 sentences away) the computed location for inter-segmental break. If so, it will declare the paragraph break to be the inter-segmental break.

5.4 Evaluation

To evaluate the identification algorithm, the procedures are applied to the ten judgment texts. The results of the automatic procedures (i.e. **Auto** column) are compared with those judged by human (i.e. **Human** column). The deviation from human judgment is reported in the **Dev** column of Table 2 and 3.

	Sentence after OP-FE Break			***Sentence before FE-RE Break***		
File	**Human**	**Auto**	**Dev**	**Human**	**Auto**	**Dev**
F01	7	7	0	97	97	0
F02	3	3	0	12	12	0
F03	3	3	0	30	9	21
F04	3	12	9	6	Fail	--
F05	3	3	0	19	8	11
F06	6	6	0	53	13	40
F07	15	11	4	27	27	0
F08	7	7	0	28	20	8
F09	6	6	0	17	9	8
F10	4	4	0	10	9	1

Table 2. Identification of *OP-FE* and *FE-RE* breaks

	RE-VD Break		
File	**Human**	**Auto**	**Dev**
F01	138	152	14
F02	43	43	0
F03	93	93	0
F04	20	20	0
F05	53	53	0
F06	116	113	3
F07	50	48	2
F08	161	161	0
F09	116	116	0
F10	30	30	0

Table 3. Identification of *OP-FE* break

As a preliminary study, the results have been encouraging. In general, the identification of *OP-FE* and *RE-VD* breaks are far better than *FE-RE* break. The errors are mostly due to occurrence of ambiguous words in *FE* segment, which affects LD reliability. For example, the word 要求 (*requisition*) can be a formal legal term or a common word for *request*. In F03, the program mistakes the occurrence of 要求 (*request*) as a legal term, affecting the identification of the end of *FE*. In F04, the small number of sentences in the text results in poor clustering, and leads to failure of identification.

6 Further Work

There are many ways to improve the system:

a. Scaling up Text Samples
We recognized that the test data used in the study is quite limited. More judgment texts should be examined using the procedures.

b. Refinement of Dictionaries
Lexical density is directly influenced by the word entries in the legal terminology dictionaries. Statistical variation of lexical items among segments should be studied. The dictionaries need to be expanded.

c. Modification of Lexical Density Formula
More factors may be introduced in the lexical density formula, e.g. word type/token ratio.

d. Intra-Segmental Structure
Currently, the model is not refined enough to identify the elaborate internal structure of reasoning segment e.g. sub-division of major points, "new arguments" in appeal cases, etc.

e. Addition of Other Features
While the variation of legal terminology is a prominent feature of judgment texts, other features will be explored in the determination of discourse segments. For example, cue phrases, conditionals, hypotheticals, etc.

f. Paragraph-based vs. Sentence-based
Paragraph was also used as the basic unit in pilot tests. However, the results based on paragraph were slightly worse than that based on sentence. It should be investigated further in the future.

7 Conclusion

In this paper, we have presented an algorithm for the identification of discourse segments in Chinese judgment texts. It will be used for the development of an XML-based tagger which will generate of discourse segment tags. The tool will help improve other NLP tasks such as summarization. The method capitalizes on the unique distribution patterns of legal terminology in Chinese legal judgment texts. Computationally, these significant changes in lexical density of legal terminololgy are detected using a difference metric. *K*-means clustering technique is applied to detect significant changes. Initial results indicate that the method is fairly accurate in identifying the major segments.

Acknowledgement
This study is supported in part by the Hong Kong Judiciary contract (#RCL/0794). We want to thank the Hong Kong Judiciary for providing legal judgment texts for this study.

References

M. Anderberg. 1973. *Cluster Analysis for Applications*. Academic Press.

K. Ashley. 1990. *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*. Cambridge, Massachusetts: MIT Press.

R. Barzilay and M. Elhadad. 1997. "Using Lexical Chains for Text Summarization." In *Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97)*, ACL, Madrid, Spain.

K. Cios, W. Pedrycz and R. Swiniarski. 1998. *Data Mining Methods for Knowledge Discovery*. Boston: Kluwer Academic Publishers.

Clark, J. and S. DeRose. 1999. XML Path Language (XPath). Version 1.0. W3C Recommendation. http://www.w3.org/TR/xpath.

O. Y. Kwong, B. K. Tsou, T. B. Y. Lai, R. W. P. Luk, L. Y. L. Cheung and F. C. Y. Chik. (2001) A Bilingual Corpus in the Legal Domain and its Applications. Workshop on Language Resources in Asia, 6th Natural Language Processing Pacific Rim Symposium, Tokyo.

A. Le Hors, P. Le Hégaret, G. Nicol, L. Wood, M. Champion and S. Byrne. (2001) Document Object Model (DOM) Level 3 Core Specification. Version 1.0. http://www.w3.org/ TR/ DOM-Level- 3-Core.

K. Sparck-Jones. 1999. "Automatic Summarising: Factors and Directions." In Mani and Maybury, (eds), *Advances in Automatic Text Summarization.* MIT press. p. 1—14.

I. Mani and M. Maybury. (eds.) 1999. *Advances in Automatic Text Summarization*. MIT Press.

D. Marcu. 2000. *The Theory and Practice of Discourse Parsing and Summarization.* Cambridge, Mass.: MIT Press.

M. Moens. 2000. *Automatic Indexing and Abstracting of Document Texts.* Boston : Kluwer Academic Publishers.