TOC Structure Extraction from OCR-ed Books

Caihua Liu[1,⋆], Jiajun Chen[2,⋆],
Xiaofeng Zhang[2,⋆], Jie Liu[1,⋆⋆], and Yalou Huang[2]

[1] College of Information Technical Science, Nankai University, Tianjin, China 300071
[2] College of Software, Nankai University, Tianjin, China 300071
{liucaihua,chenjiajun,zhangxiaofeng}@mail.nankai.edu.cn,
{jliu,ylhuang}@nankai.edu.cn

Abstract. This paper addresses the task of extracting the table of contents (TOC) from OCR-ed books. Since the OCR process misses a lot of layout and structural information, it is incapable of enabling navigation experience. A TOC is needed to provide a convenient and quick way to locate the content of interest. In this paper, we propose a hybrid method to extract TOC, which is composed of rule-based method and SVM-based method. The rule-based method mainly focuses on discovering the TOC from the books *with* TOC pages while the SVM-based method is employed to handle with the books *without* TOC pages. Experimental results indicate that the proposed methods obtain comparable performance against the other participants of the ICDAR 2011 Book structure extraction competition.

Keywords: table of contents, book structure extraction, xml extraction.

1 Introduction

Nowadays many libraries focus on converting the whole libraries by digitizing books on an industrial scale and this project is referred as '*digital libraries*'. One of the most important tasks in digital libraries is extracting the TOC. A table of contents (TOC) is a list of TOC entries each of which consists of three elements: title, page number and level of the title. The intention of extracting TOC is to provide a convenient and quick way to locate content of interest. To extract the TOC of the books, we are faced with several challenges. First, the books are various in forms, since the books come from different fields and there are kinds of layout formats. A large variety of books increases the difficulty of utilizing a uniform method to well extract the TOC. Taking the poems for example, some poems contain TOC pages while some not. The alignment of poems may be left-aligned, middle-aligned and right-aligned. Second, due to the limitation of OCR technologies, there are a certain number of mistakes. OCR mistakes also cause trouble in extracting TOC, especially when some keywords such as 'chapters', 'sections', etc., are mistakenly recognized.

⋆ The first three authors make equal contributions.
⋆⋆ Corresponding author.

S. Geva, J. Kamps, and R. Schenkel (Eds.): INEX 2011, LNCS 7424, pp. 98–108, 2012.

Many methods have been proposed to extract TOC, most of which are published in INEX[1] workshop. Since 80% of these books contain table of contents, MDCS [1] and Noopsis [2] took the books with table of content into consideration. While the University of Caen [3] utilized a four pages window to detect the large whitespace, which is considered as the beginning or ending of chapters. XRCE [4] segmented TOC pages into TOC entries and used the references to obtain page numbers. XRCE also proposed a method trailing whitespace.

In this paper, we propose a hybrid method to extract TOC, since there are two types of data. 80% of the books contain TOC pages and the remaining do not. This two situations are considered via rule-based method and SVM-based method respectively. For books containing TOC pages, some rules are designed to extract these TOC entries. The rules designed are compatible with the patterns of most books, which is also demonstrated in the experiments. For books without TOC pages, a SVM model is trained to judge whether one paragraph is a title or not. A set of features is devised for representing each paragraph in the book. These features also do not depend on knowledge of TOC pages. Using these features and the machine learning method we can extract TOC entries, whether the book has an TOC page or not. To better organize these TOC entries, the level and the page number of each TOC entry locating are also extracted, besides these TOC entries themselves.

The paper is organized as follows. In section 2, we give a description of the previous works about the extraction of TOC. An introduction about the books and the format of these data is presented in section 3. The main idea of our works to extract TOC entries is shown in section 4. In section 5, we locate the target page for each TOC entry. We assign levels for TOC entries in section 6. Finally, experiments and a short conclusion are displayed.

2 Related Works

In the application of digital libraries, there are four main technologies, information collecting, organizing, retrieving and security. Organizing data with XML is the normal scheme, especially when we are faced with large scales of data. A information retrieval workshop named INEX has been organized to retrieve information from XML data. In 2008, BSE(Book Structure extraction) was added to INEX, whose purpose is to evaluate the performance of automatic TOC structure extraction from OCR-ed books.

MDCS [1] and Noopsis[2] focus on books containing TOC pages. Except for locating the TOC entries, they make no use of the rest of the books. MDCS employees three steps to extract TOC entries. Firstly, they recognize TOC pages. Secondly, they assign a physical page number for every page. Finally, they extract the TOC entries via a supervised method relying on pattern occurrences detected. MDCS's method depends on the TOC pages and it can not work for books without TOC pages. University of Caen's [3] method did not rely on the content page, the key hypothesis of which is that the large whitespace is the

[1] `http://www.inex.otago.ac.nz/`

beginning of one chapter and the ending of another chapter. So they utilized a four pages window to detect the whitespace. However, it works well on high-level title but the lower title can not be recognized well. Xerox Research Center Europe [4] used four methods to extract the TOC entries. First two are based on TOC pages and index pages. The third method is similar to MDCS, which also defines some patterns while the last method trails the whitespace like University of Caen does. The method proposed by us is more efficient than others, since we directly extract TOC entries by analyzing the TOC pages for books with TOC pages. Due to the diversity of books without TOC pages, it's difficult to find a uniform rule or pattern to extract the TOC entries. To these issues, we perform an automatic learning method for extracting TOC entries.

3 The Architecture of the Hybrid Extracting Method

In this section, we will give a description of the architecture of our method. Since 80% of the books contain TOC pages while the remaining do not, we consider these two situations respectively. One more crucial reason is that the TOC pages contain a lot of hidden structural contents. The well using of the TOC pages can help improve the extracting performance. In addition, for books with TOC pages, directly extracting TOC from the TOC pages performs better than extracting TOC from main text.

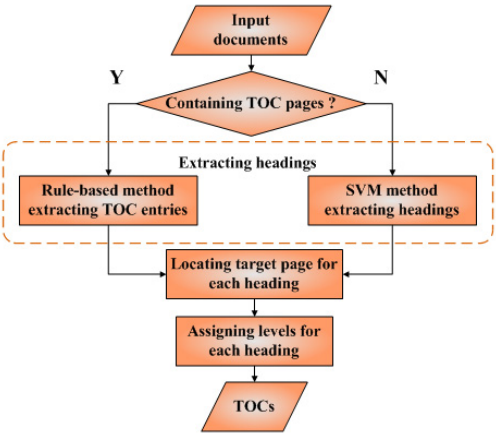


Fig. 1. The flow chart of our hybrid extracting system

As previously stated, each TOC entry contains three parts: title, page number and level of the title. As shown in Figure 1, the extracting process is conducted with the following steps. (1) A judgement is conducted to separate the books into two parts via whether containing TOC pages. (2) Extracting each TOC

entry and obtaining the title, page number. Since there are two types of data, we extract the TOC from them respectively. For books containing TOC pages, a rule-based method is proposed to extract these TOC entries from the TOC pages. For books without TOC pages, a SVM method is introduced to achieve the purpose of extracting TOC. (3) Locating the target page for each TOC entry. (4) Assigning levels for these extracted TOC entries. A simple relationship between these TOC entries can be obtained by this step.

After these steps, each TOC entry has been extracted. we also obtain the target pages and the organizational structure of these TOC entries. Then the TOC is outputted as the predefined format. Until now, all of the works to extract TOC has been accomplished. And the specific methods to conduct this three steps is stated in the following sections.

4 Extracting TOC Entries

In this section, we focus on the two methods to extract TOC entries. The following two sections will give a specific introduction of these two methods respectively.

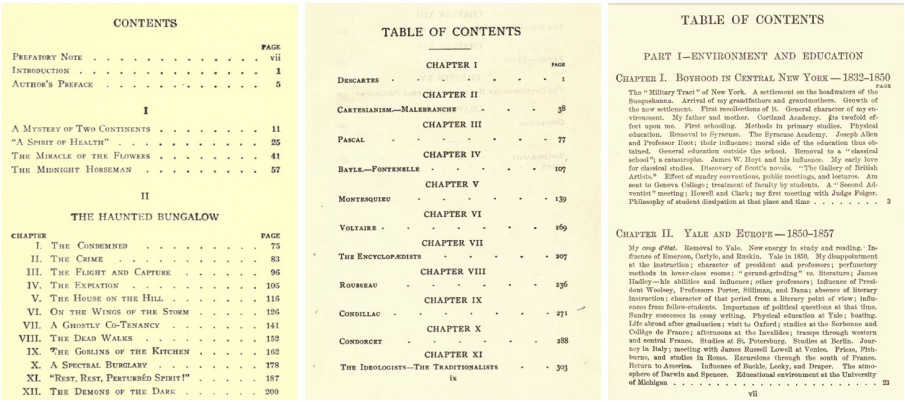
4.1 Extracting with TOC Pages

We extract the TOC from the original TOC page in the book with TOC pages. This task is divided into two steps: locating the TOC pages of the book and extracting TOC entries from the TOC pages.

Locating TOC Pages. If a book contains TOC pages, we extract the TOC entries from the TOC pages directly. Naturally, the TOC pages start with key words such like '*Contents*' and '*Index*', and the TOC page contains many lines ending with numbers. We can use these features to locate the beginning of contents pages. Since most books have headers, the pages are ascertained to be TOC pages, if there are key words like '*Contents*' and '*Index*' appearing at the beginning of the page, or if there is a considerable number of lines ending with numbers. Naturally, TOC pages usually appear in the front part of a book, as a result, we only need to consider the first half of the book to accelerate the process.

Extracting Contents Entries. TOC entries in books vary greatly in different books. As is shown in Figure 2, some entries like Figure 2(a) occupy only one line, while some entries Figure 2(b) occupy several lines. Some entries Figure 2(c) are divided into multi-lines, of which the first line is text, the second line is logical page number and the third line is introduction of the section.

To extract each TOC entry, we need to obtain the beginning and the ending of each TOC entry. The following rules are conducted to identify the beginning of TOC entries.



CONTENTS

PAGE
PREFATORY NOTE vii
INTRODUCTION 1
AUTHOR'S PREFACE 5

I

A MYSTERY OF TWO CONTINENTS 11
"A SPIRIT OF HEALTH" 25
THE MIRACLE OF THE FLOWERS 41
THE MIDNIGHT HORSEMAN 57

II

THE HAUNTED BUNGALOW

CHAPTER PAGE
I. THE CONDEMNED 75
II. THE CRIME 83
III. THE FLIGHT AND CAPTURE 96
IV. THE EXPIATION 105
V. THE HOUSE ON THE HILL 116
VI. ON THE WINGS OF THE STORM 126
VII. A GHOSTLY CO-TENANCY 141
VIII. THE DEAD WALKS 152
IX. THE GOBLINS OF THE KITCHEN 162
X. A SPECTRAL BURGLARY 178
XI. "REST, REST, PERTURBÉD SPIRIT!" 187
XII. THE DEMONS OF THE DARK 200

TABLE OF CONTENTS

CHAPTER I PAGE
DESCARTES 1
CHAPTER II
CARTESIANISM.—MALEBRANCHE 38
CHAPTER III
PASCAL 77
CHAPTER IV
BAYLE.—FONTENELLE 107
CHAPTER V
MONTESQUIEU 139
CHAPTER VI
VOLTAIRE 169
CHAPTER VII
THE ENCYCLOPÆDISTS 207
CHAPTER VIII
ROUSSEAU 236
CHAPTER IX
CONDILLAC 271
CHAPTER X
CONDORCET 288
CHAPTER XI
THE IDEOLOGISTS—THE TRADITIONALISTS . . . 303
ix

TABLE OF CONTENTS

PART I—ENVIRONMENT AND EDUCATION

CHAPTER I. BOYHOOD IN CENTRAL NEW YORK—1832–1850
PAGE
The "Military Tract" of New York. A settlement on the headwaters of the Susquehanna. Arrival of my grandfathers and grandmothers. Growth of the new settlement. First recollections of it. General character of my environment. My father and mother. Cortland Academy. Its twofold effect upon me. First schooling. Methods in primary studies. Physical education. Removal to Syracuse. The Syracuse Academy. Joseph Allen and Professor Hoot; their influence; moral side of the education thus obtained. General education outside the school. Removal to a "classical school"; a catastrophe. James W. Hoyt and his influence. My early love for classical studies. Discovery of Scott's novels. "The Gallery of British Artists." Effect of sundry conventions, public meetings, and lectures. Am sent to Geneva College; treatment of faculty by students. A "Second Adventist" meeting; Howell and Clark; my first meeting with Judge Folger. Philosophy of student dissipation at that place and time 3

CHAPTER II. YALE AND EUROPE—1850–1857

My coup d'état. Removal to Yale. New energy in study and reading. Influence of Emerson, Carlyle, and Ruskin. Yale in 1850. My disparagement at the instruction; character of president and professors; perfunctory methods in lower-class rooms; "ground-grinding" vs. literature; James Hadley—his abilities and influence; other professors; influence of President Woolsey, Professors Porter, Silliman, and Dana; absence of literary instruction; character of that period from a literary point of view; influences from fellow-students. Importance of political questions at that time. Sundry successes in essay writing. Physical education at Yale; boating. Life abroad after graduation; visit to Oxford; studies at the Sorbonne and Collège de France; afternoons at the Invalides; tramps through western and central France. Studies at St. Petersburg. Studies at Berlin. Journey in Italy; meeting with James Russell Lowell at Venice. Prices, Pitchburns, and studies in Rome. Excursions through the south of France. Return to America. Influence of Buckle, Lecky, and Draper. The atmosphere of Darwin and Spencer. Educational environment at the University of Michigan 23

vii

Fig. 2. A variety of Contents structure. (a) Each TOC entry occupy one line; (b) Each TOC entry occupy several lines; (c) Each TOC entry occupy multi-lines.

1. if current line starts with key words like '*Chapter*', '*Part*', '*Volume*' or '*Book*' etc.
2. if current line starts with numbers or Roman numerals.
3. if last line ends with numbers or Roman numerals.

The start of a new TOC entry is the end of last TOC entry, so we can easily construct the following rules to identity the end of TOC entries.

1. if next line starts with key words like '*Chapter*', '*Part*', '*Volume*' or '*Book*' etc.
2. if next line starts with numbers or Roman numerals.
3. if current line ends with numbers or Roman numerals.

The above rules can handle most of the situations, however, some TOC entries such as multi-lines can not be well extracted using only those rules. To these issues, a new rule is added. If the last line does not have key words like '*Chapter*' and Roman numerals etc. that obviously separate contents items and the formats of current line and last line are very different, delete line information collected before.

The new rule treats current line as the start of a new TOC entry, and delete stored lines formerly should be treated as part of current TOC entry. The difficulty of this rule lies in the quantitative description of differences between two lines. Our approach only consider the relative_font_size.

$$relative\_font\_size = \frac{currentlinefontsize - averagefontsize}{maximumfontsize - avergefontsize} \quad (1)$$

Where the value of font_size is the average height of words in one line, and the height is computed by the position of words. If the '*relative_font_size*' of the two lines are very different and greater than some pre-set thresholds, these two lines will not be treated as one TOC entries.

4.2 Extracting without TOC Pages

Considering of establishing a uniform model for all the two types of books, we conduct a automatic method to label training data in the favor of TOC. It is expected that the SVM[2] model can handle both this two types data well. However, the method performs worse than the rule based method, so the SVM method is only utilized on books without TOC pages. It comes into the following steps: (1) extracting the features of each paragraph and labeling them(2) training the RBF-SVM to classify every paragraph.

Features. Through observing data set, some obvious features can be employed to identify a heading, as shown in table 1. Though we get the eight features, it happens that some common paragraphs have one or more features. For example, the page header is much more similar to the heading, so this will confuse the classifier. Commonly the page header always has the same content with the title, while it has a lot of duplications. So we use post-process method to delete the duplication and make the first page header that has the same content with others as the title. Another example, the title page (this page only contains a title, and in the next page it also starts with this title in the top) has also some of the features, what's more the effect of the features is more obvious than the title at times. So we must do some efforts to solve the problem. According to the title page, we set a threshold to judge whether a page is title page. It is means that if most of the paragraphs in the page are recognized as title, we think it is a title page.

Table 1. Features designed for books without TOC pages

Feature ID	Discription
1	Proportion of Capital Letters
2	Font Size
3	Left End position of a Paragraph
4	Right End Position of a Paragraph
5	Space between Paragraph
6	Line Number of a Paragraph
7	Average Number of words in Each Line of a Paragraph
8	The y-coordinate of a Paragraph Start

Recognizing the TOC Entries. We use SVM to identify whether one paragraph is TOC entry. Since many normal paragraphs are predicted as positive, a further analysis is made on these data. There are four situations to consider: the title we expect, page header, the misrecognized paragraph and the spot or handwritten note in the book which can be OCRed. In order to get a much higher performance, a post-process is conducted. And the following principles are devised to delete some of the positive ones.

[2] `http://www.cs.cornell.edu/people/tj/svm_light.html`

1. If there are less capital letters in a paragraph than a threshold.
2. If a paragraph only contains a letter, and it is not Roman number as well.
3. If a paragraph is similar to others,then keep the first one and delete the others.
4. If there are more than two positive paragraphs(paragraphs predicted by SVM as headings).

5 Locating Target Page

After extracting TOC entries, we need to ascertain where the entries actually locate for navigation purpose. The page numbers shown in TOC are the logical numbers. While the physical number shows the actual page number in the whole document. Hence the matching of physical page number and logical page number is expected to help users navigate over the whole document.

To match the physical and the logical page numbers, the logical numbers for every page are needed to be extracted first. Commonly, the logical number appears in the headers or footers. However, logical pages extracted in this way are not prefect enough, as some pages may indeed do not have logical page numbers or maybe an OCR error makes the logical page numbers not recognized correctly, so we need to deal with those omissions and errors. So a remedy is conducted to obtain the complete page numbers. First, fill the vacancies of pages without page numbers using the following method. If the physical page i and j ($j > i$) have logical page $L(i)$ and $L(j)$ respectively, and logical page numbers of pages between i and j are absent, at the same time, if $L(j) - L(i) = j - i$, fill the vacancies of logical page numbers for those page between i and j.

Whereas, there are still some pages with physical numbers which can not find logical number. For these physical pages we set them to 0. And then the logical page numbers of the extracted contents items are replaced to physical page numbers. For these physical pages labeled '0', We first find the the maximum logical page number smaller then current logical page number, and match with physical page number. Then from this maximum logical page number and forward we use text match to find the first page that contains the text of current TOC entry, and use the physical page number of this page as the physical page of current TOC entry.

6 Ranking Levels for TOC Entries

The final step is to rank those extracted TOC entries. Via the analysis of the data, we find that most of the content entries contain the key term '*Book*','*Volume*','*Part*', '*Chapter*' and so on. While information like arabic numerals and roman numerals can also be utilized to assign the level for content entries. So we pre-define five levels to arrange the levels of every contents entry.

1. First level: containing key words '*Part*', '*Volume*' and '*Book*' etc.
2. Second level: containing keywords '*Chapter*' and '*Chap*' etc.
3. Third level: containing keywords '*Section*' and '*Sect*' etc.
4. Forth level: containing Arabic numerals and Roman numerals or keywords like '*(a)*' and '*a*'.
5. To be ascertained level: other TOC entries that do not have above mentioned features while its level depends on their neighbors' contents, for example, previous rank.

A specific statistics on randomly selecting 100 books from the ICDAR 2011 dataset is shown in Figure 3. 70% of TOC entries contain keywords 'chapter' etc. and books with keyword 'Section' only occupy a small proportion of 100 books. More obvious is that 90% of the books correspond to our definition of levels. We first scan the whole content entries and assign levels for every entry

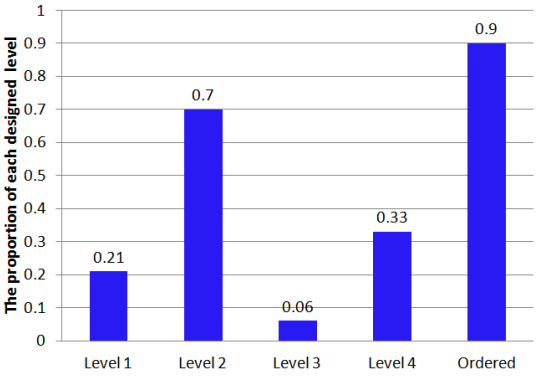


Fig. 3. The percent of each level and 'Ordered' means that the level of TOC corresponds to the level we defined

by the rule pre-defined above. Most of these entries are all assigned levels, only these entries without any characteristics left. A statistics has been conducted by us and it demonstrates that these left entries have a higher probability of the same level as the previous one, therefore, the levels for these left entries are assigned via this idea.

7 Experiments

In order to measure the performance of our method, we conduct experiment on two datasets. One is the 1000 books provided by the Book structure extraction competition, while the other is ICDAR 2009 competition dataset. The pdf and DjVuXML format of the books are both provided in these two datasets.

To give a full evaluation of the performance, three evaluate criterions are considered on five aspects. The evaluation measures are: precision, recall and

F-measure. And the 5 aspects are: (1) Titles, which evaluates whether the titles we obtained are sufficiently similar to the titles in the ground truth; (2) Links, which is correctly recognized if the TOC entries recognized by our method link to the same physical page in the ground truth. (3) Levels, which means whether established level of the title is at the same depth in the ground truth. (4) Complete except depth, which represents the title and the link are both right. (5) Complete entries, which is considered right only when all of these three items are right.

7.1 Experiments on the Whole Dataset

The experiments listed in this section are the public experimental results published by the official organizing committee of ICDAR. The performance of our method on the five aspects are reported in table 2. The performance comparison between our method with other participants of ICDAR is presented in Figure 4. It can be seen that MDCS outperforms others, but it is a TOC based method and it can not deal with books without table well. We rank second in the completion, however, we are capable of processing those books without contents. To address these issues, our method is comparable to others.

Table 2. The performance of our method on five evaluation aspects conducting on ICDAR 2011 dataset

Items	Precision	Recall	F-Measure
Titles	47.99%	45.70%	45.20%
Links	44.03%	41.44%	41.43%
Level	37.91%	36.84%	36.08%
Entries disregarding depth	44.03%	41.44%	41.43%
Complete entries	34.80%	33.28%	33.06%

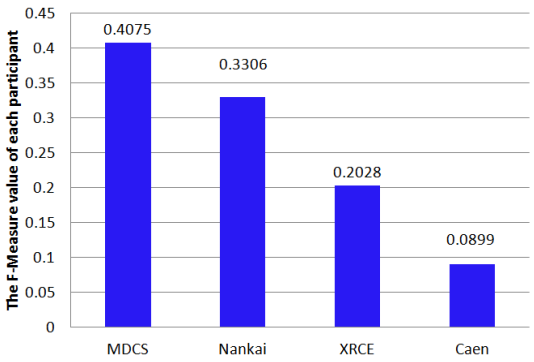


Fig. 4. The public results of ICDAR 2011 book structure extraction competition. It is conducted on the whole dataset and we rank second.

7.2 Experiments on Books without Table of Contents

To evaluate the ability of processing books without TOC pages, we conduct experiments on books without TOC pages in 2009 and 2011 ICDAR dataset respectively. The training data for SVM is obtained from the books with TOC pages and we try to learn how TOC entries look like. Since the number of normal context is much larger than the number of TOC entries, so we use all of the TOC entries and randomly select the same number of normal text.

Table 3. The performance of our method on five evaluation aspects conducting on ICDAR 2009 dataset

Items	Precision	Recall	F-Measure
Titles	14.85%	23.64%	14.38%
Links	10.88%	16.17%	10.71%
Level	11.78%	19.62%	11.40%
Entries disregarding depth	10.88%	16.17%	10.71%
Complete entries	8.47%	13.07%	8.43%

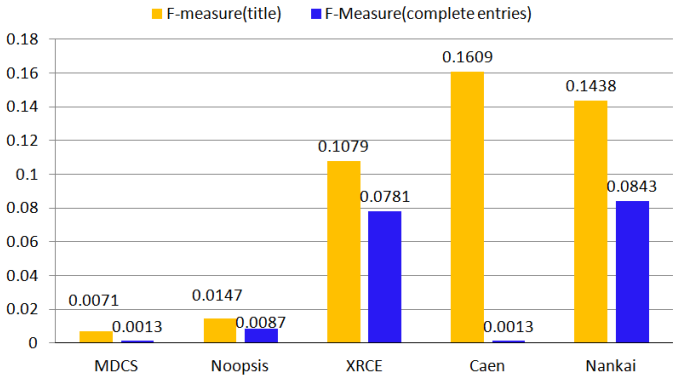


Fig. 5. Experiment on books without TOC pages and it is conducted on 93 books of ICDAR 2009 dataset

The F-Measure of complete entries on the 2011 ICDAR dataset is 5.20%. Owing to no result of books without table of contents is publicly published, we also conduct our experiment with the 2009 ICDAR dataset to make a comparison. The result of five fold-cross validation on 2009 dataset is shown in Table 3. All of these five evaluate aspects are considered to give a full description of our method. Figure 5 shows the comparison with other methods public published in ICDAR 2009. It can be seen that our method outperforms others'.

8 Conclusion

This paper presents the task of extracting TOC entries for navigation purpose. Due to the missing of layout and structural information caused by the OCR process, how to extract TOC entries from OCR-ed books becomes a challenging problem. We proposed an effective method to solve this problem. For books containing contents page, a rule based method is conducted. For books without contents page, we utilize a machine learning method to classify the title. Besides, recognition of the title, the matching of physical and logical page is conducted to help users navigate. To get more specific information about the book, the partition of the level of title is also performed. The experiments show that our method considering these three aspects is usable and effective. A uniform model to effectively address the problem is expected as a future work.

Acknowledgments. This research was supported supported by the Fundamental Research Funds for the Central Universities and the National Natural Science Foundation of China under Grant No. 61105049.

References

1. Dresevic, B., Uzelac, A., Radakovic, B., Todic, N.: Book Layout Analysis: TOC Structure Extraction Engine. In: Geva, S., Kamps, J., Trotman, A. (eds.) INEX 2008. LNCS, vol. 5631, pp. 164–171. Springer, Heidelberg (2009)
2. Doucet, A., Kazai, G., Dresevic, B., Uzelac, A., Radakovic, B., Todic, N.: Setting up a Competition Framework for the Evaluation of Structure Extraction from OCR-ed Books. International Journal of Document Analysis and Recognition (IJDAR) 14(1), 45–52 (2010)
3. Giguet, E., Lucas, N.: The Book Structure Extraction Competition with the Resurgence Software at Caen University. In: Geva, S., Kamps, J., Trotman, A. (eds.) INEX 2009. LNCS, vol. 6203, pp. 170–178. Springer, Heidelberg (2010)
4. Déjean, H., Meunier, J.-L.: XRCE Participation to the 2009 Book Structure Task. In: Geva, S., Kamps, J., Trotman, A. (eds.) INEX 2009. LNCS, vol. 6203, pp. 160–169. Springer, Heidelberg (2010)