An Improved Algorithm of Logical Structure Reconstruction for Re-flowable Document Understanding

Lin Zhao, Ning Li(✉), Xin Peng, and Qi Liang

Department of Computer, Beijing Information Science and Technology University, Beijing, China
zhaolin_0124@126.com, ningli.ok@163.com

Abstract. The basic idea of re-flowable document understanding and automatic typesetting is to generate logical documents by judging the hierarchical relationship of physical units and logical tags based on the identification of logical paragraph tags in re-flowable document. In order to overcome the shortages of conventional logical structure reconstruction methods, a novel logical structure reconstruction method of re-flowable document based on directed graph is proposed in this paper. This method extracts the logical structure from the template document and then utilizes directed graph's single-source shortest path algorithm to filter out redundant logical tags, thus solving the problem of logical structure reconstruction of a document. Experimental results show that the algorithm can effectively improve the accuracy of logical structure recognition.

Keywords: Logical structure reconstruction · Document understanding · Logical tags

1 Introduction

With the popularity of electronic documents, re-flowable document has been used more and more extensively. Previous format documents and the most widely used re-flowable documents at present have to deal with logical structure reconstruction of the document [1-2]. For format document (Figure 1), the region and positional relationship of diagrams, graphics, tables and text information are first cut apart and judged automatically through layout analysis, and then the physical geometry structure will be mapped to logical structure through layout understanding. But re-flowable document does not contain high-level logical structure. The high-level structure needs to be deduced from the low-level elements (Figure 2).

So if the information structure of the re-flowable document can be accurately identified, it will have a great significance for document understanding, especially for the applications such as document retrieval and format check of document. Chinese and international researchers have carried out studies on of logical structure reconstruction of documents, including the method based on rules (i.e. the methods are proposed by LeBourgeois [3], Rosenfeld [4], Hu [5]), and the method based on statistics (i.e. Brugger [6] and Palmero [7]), the method based on FSA(i.e. Song Haosu [8]).

J. Li et al. (Eds.): NLPCC 2015, LNAI 9362, pp. 339–346, 2015.
DOI: 10.1007/978-3-319-25207-0_28

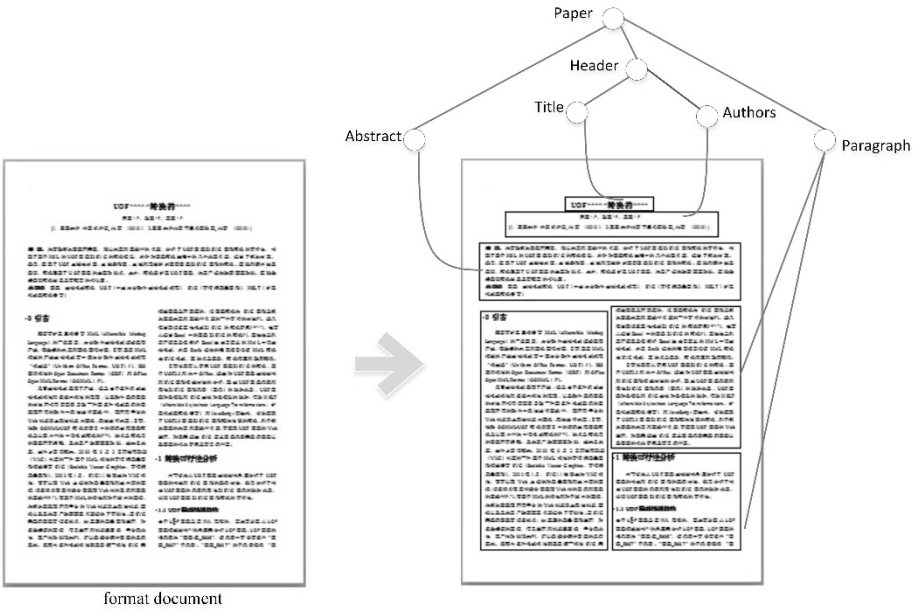

Fig. 1. Format document understanding

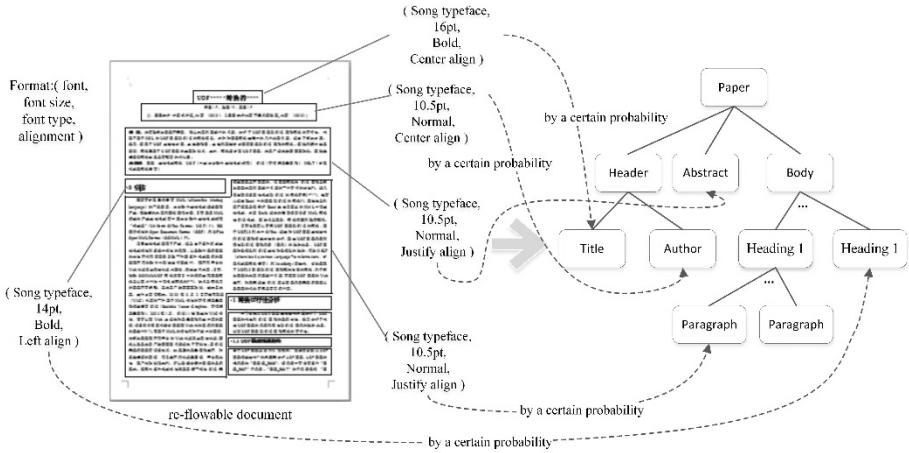

Fig. 2. Logical structure reconstruction in a re-flowable document

Rule-based reconstruction method is an iterative algorithm, which tries to find out an optimal global solution based on text features. Statistics-based reconstruction method requires a lot of training sets and is more dependent on the training text corpus. FSA-based reconstruction method, may include some states containing errors due to possible typesetting errors, so they can result in an infinite state if added to the FSA. Based on the above situation, a directed graph-based logical structure

reconstruction method for documents on the basis of FSA logical structure reconstruction method is presented in this paper.

2 The Emergence of Logical Structure Reconstruction Method Based on Directed Graph

2.1 Logical Tag Structure Extraction from a Template Document

Logical tags, such as “CAT”, “CAB”, etc., represent the logical roles of paragraphs in a document. Logical structure reconstruction of a document needs a reconstruction rule which is determined by extracting logical tag structure from template document.

Extracting logical tag structure is to determine the sequence of logical document tags [9], i.e. the sequence of document roles, which provides a reconstruction basis for logical structure reconstruction of a document. To take thesis as an example, assuming the thesis with a maximum of Heading 3, then the sequence of logical structure tags extracted from the template document is shown in Figure 3.

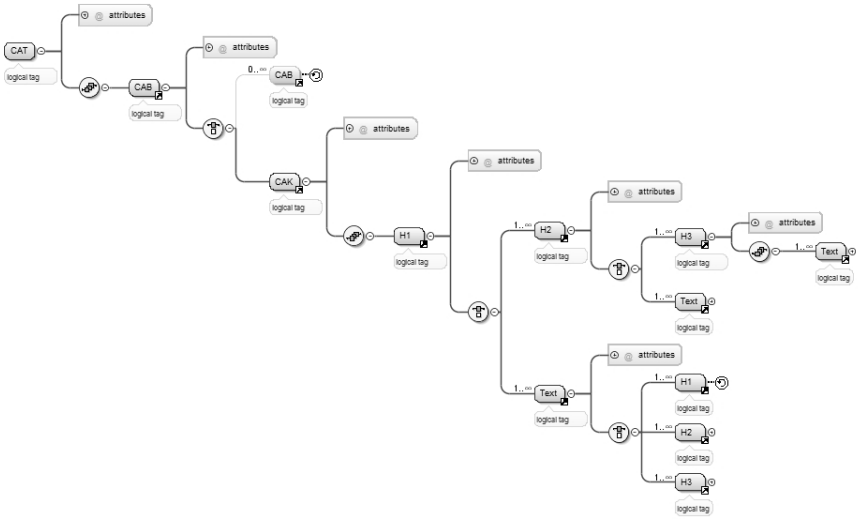

Fig. 3. Hierarchical structure of logical document tags

The meanings of the elements in the Figure 3 are as follows. CAT(Chinese Abstract Title), CAB(Chinese Abstract Body), CAK(Chinese Abstract Keywords), H1 (Heading 1), H2(Heading 2), H3(Heading 3), Text(Normal(body)). The attributes in the Figure 3 are shown in Figure 4.

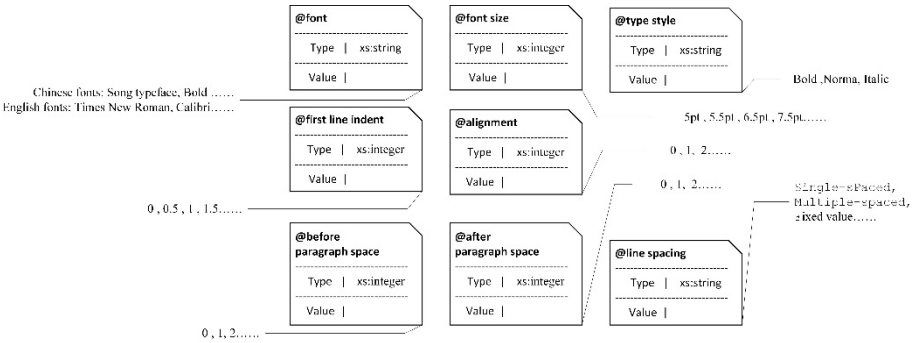

Fig. 4. The attributes in the Figure 3

2.2 Construction Method of Directed Graph of a Document

The logical tags of a paragraph shall be first identified when constructing nodes of the directed graph [10]. Since some logical structures have similar format, misjudgment of logical tags may occur. Therefore, logical tags need to be filtered according to the logical tag structure extracted from section 2.1. The impossible logical tags of the paragraph need to be filtered out to make preparations for further implementation of the logical structure reconstruction. Accordingly, the construction of the directed graph nodes can be divided into the following two steps:

Determination of Logical Tags

By section 2.1, every paragraph in a document corresponds to a format vector which has eight components. Then the to-be-investigated document dictionary in which the values of the components of the to-be-investigated document are recorded and the template document dictionary in which the values of the components of the template document are recorded are then established. Every value corresponds to a different branch which is composed of similar values and the paragraph which contains the value.

For the paragraph which needs logical tag identification, the score of each logical tag in the paragraph is calculated and the tag with highest score is most likely the logical tag of the paragraph. If there are two highest scores, they should be seen as candidate logical tags and will be further filtered in latter selection process. The score of a logical tag of a paragraph is the sum of the scores of its logical components of this paragraph format vector. The score of a logical tag equals the degree of similarity of the branch times the number of occurrences of the logical tag in the branch times the weight of the component of the logical tag. The weight value of the component is evaluated by the frequency of occurrence of each component's value in the document format, the higher the frequency of occurrence, the greater the corresponding weight. The basic algorithm of logical tag identification is shown as follows:

```
1: for p in P do
2:   for l in L do
3:     for c in C do
```

```
4:        for d_w in dic_c do
5:        if F_c(p)= d_w && s(d_w,w) = max{s(d_w,d_w's branches)}
6:          then score_c(l) = s(w,d_w)*e(l,d_w's index)*w(l,c)
7: score(l) = Σ_{c∈C} score_c(l)
8: if score(l_i) = max(score(l)) then
9:   return l_i
```

where s (d_w, w) is the degree of similarity membership degree of the vocabulary d_w of the to-be-investigated document and the vocabulary w of the template document. e(l, d_w's index) is the number of occurrences of the logical tag l in the vocabulary d_w. w(l, c) is the weight of the logical tag l on the component c, $score_c$(l) is the score of the logical tag l on the component c, score(l) is the total score of the logical tag l on all the components.

Filtering of Logical Tags

Logical tag filtering has adopted the same approach. Logical tags are filtered by combing the logical tags of the previous paragraph with the logical tags of the template. By default, the logical tag of the first paragraph shall be retained and the subsequent paragraphs will be examined one by one. The logical tags left by the previous paragraph shall be checked if it is the expected tag of the current paragraph according to the sequence of logical tags in the template. If it is, then the logical label shall be retained. If not, then the tag shall be filtered out. If no result is the expected tag of the current paragraph, then all the judgment results of the previous paragraph shall be filter out, and the logical tags of the current paragraph shall be kept. However, the logical tag is not unique. After logical tag filtering, some paragraphs may have only one tag, other paragraphs may have multiple logical tags due to multiple logical tag judgment results. For this situation, probability of occurrence of a paragraph shall be added, it will be detailed introduction in section 2.3.

The nodes of the directed graph were obtained from previous paragraphs. The edges of the directed graph are used to connect each node so as to show the relationship of the nodes. The logical tag left by each paragraph is the corresponding tag of the paragraph in the graph, and each node of the previous paragraph points to every node of the next paragraph.

2.3 Reconstruction Method of the Logical Structure of a Document

The single-source shortest-path algorithm describes a way to find out the shortest path from a certain source point s ∈ V to the rest vertices in V in a known directed-weighted graph G = (V, E). In the graph G, the weighting function w is defined, w:E → R is the mapping from the edge to real weight value. We adopt Bellman-Ford algorithm [11] here. The shortest path from u to v is defined as:

$$\delta(u,v)=\begin{cases}\min\{w(p):u\to v\} & a\ pathway\ from\ u\ to\ v\\ \infty & else\end{cases} \tag{1}$$

At the beginning, the path length of the source point u is 0. Meanwhile, the path lengths of all other vertices are assigned as ∞, which indicates that the paths going to all these vertices (except u and v) are unknown. After the continuous updating of values, the shortest sequence which is the shortest path from u to v will be obtained eventually. The basic idea is as follows:

```
1: for i ← 1 to |V[G]| - 1 do
2:   for EDGE (u, v) in E[G] do
3:     if d[v] > d[u] * 1 / w(u, v)
4:       then d[v] ← d[u] * 1 / w(u, v)
5:      [v] ← u
6: for EDGE (u, v) in E[G]  do
7:   if d[v] > d[u] + w(u, v)
8:     then return FALSE
9: return TRUE
```

The weight of the path is the sum of all weight values of its edges in this algorithm. In order to be more suitable for the logical structure reconstruction of a document, we redefine the path length. The path length is the reciprocal of the probability of existence of the path, the smaller the length is, the bigger the probability of existence of the path is, the bigger the possibility of the corresponding logical structure is. In this algorithm, the calculation of the weight on the graph edge is as follows: if the starting point of the edge is l_i and the ending point of the edge is l_j, then the weight w (l_i, l_j) of the edge is equal to the reciprocal of the probability of occurrence P (l_i, l_j) of corresponding logical tag of the starting point and ending point. Then the number of occurrences of various logical tags after each tag is calculated. C (l_i, l_j) is the number of occurrences of l_j after l_i, C(l_i) is the total number of occurrences of all logical tags after li, then the probability of occurrence of l_j after l_i is P (l_i, l_j) = C (l_i, l_j) / C (l_i). Then the single-source shortest path algorithm can thus be obtained. Since there may be multiple shortest paths, there may exist more than one corresponding logical document structure.

3 Experiment and Related Analysis

The corpus used by the experimental set is built by our university. We extract 500 documents from the corpus, which are the theses of undergraduate students including but not limited to “CAT”, “CAK”, “H1”, “H2”, “H3”, “Text” and then carry out artificial identification and labeling on their paragraphs human. And then we select 200 documents randomly to set up learning sample set and the rest 300 documents are used to establish the test sample set. For the learning sample set, the logical structure reconstruction method based on directed graph is applied to do statistics. Then this method is applied to the test sample set which is reconstructed using the logical structure reconstruction of a document to collect results. The evaluation criterion of logical structure reconstruction of a document is the proportion of the paragraphs whose logical tags are determined by the logical structure reconstruction method in all

the paragraphs. Precision, Recall and Balance F value are used to evaluate the evaluation criterion of the logical paragraph tags. It is defined as follows:

$$P = \frac{|\{relevant\ \ paragraphs\} \cap \{retrieved\ \ paragraphs\}|}{|\{retrieved\ \ paragraphs\}|} \tag{2}$$

$$R = \frac{|\{relevant\ \ paragraphs\} \cap \{retrieved\ \ paragraphs\}|}{|\{relevant\ \ paragraphs\}|} \tag{3}$$

$$F = \frac{2 \times P \times R}{P + R} \tag{4}$$

For the identification method of logical tags here we use the method mentioned in section 2.3 and logical tags identification method based on VSM. The identification results are shown in Table 1.

Table 1. Identification results of logical tags of a paragraph before and after the improvement

Paragraph role	Before the improvement			After the improvement			F change
	P	R	F	P	R	F	
CAT	97.62%	86.52%	91.73%	96.54%	86.12%	91.03%	-0.70%
CAK	98.33%	80.64%	88.61%	98.43%	82.36%	89.68%	+1.07%
H1	95.41%	78.42%	86.08%	96.65%	79.86%	87.45%	+1.37%
H2	94.39%	79.43%	86.26%	95.01%	79.23%	86.40%	+0.14%
H3	94.47%	76.28%	84.40%	95.40%	78.18%	85.93%	+0.53%
Text	98.54%	80.47%	88.59%	96.23%	76.89%	85.48%	-3.11%

As can be seen from the table 1, the precision and recall rate of the improved logical paragraph tag identification are high, indicating that it can accurately find the most similar logical paragraph formatting tags and can also find out a logical tag's paragraph as many as possible. We apply T-Test to P (before the improvement and after the improvement) in Table1, and the results are shown in Table 2.

Table 2. Paired Samples Test

Pair1	Paired Differences					t	df	Siq. (2-tailed)
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference				
				Lower	Upper			
before - after	8.33E-04	1.36E-02	5.55E-03	-1.34E-02	1.51E-02	0.15	5	0.887

4 Conclusions

The algorithm of logical structure reconstruction has great significance for document understanding. However, because of many factors are considered in the proposed algorithm, so there is a certain limitation in the study. Among them, it is a lack of consideration for complex structure in the logical structure reconstruction algorithm. In addition, there is no direct connection between the probability of solution and the probability of shortest path. Therefore, we still make every endeavor to resolve these problems further. In the next research work, it is an important research perspective for how to apply the recognized document information structure to document format checking.

Acknowledgement. This paper is supported by the Project of Construction of Innovative Teams and Teacher Career Development for Universities and Colleges Under Beijing Municipality (No.IDHT20130519), and the general program of science and technology development project of Beijing Municipal Education Commission (No.KM201511232013).

References

1. Mao, S., Rosenfeld, A., Kanungo, T.: Document structure analysis algorithms: a literature survey. In: Electronic Imaging 2003, International Society for Optics and Photonics, pp. 197–207 (2003)
2. Namboodiri, A.M., Jain, A.K.: Document structure and layout analysis. In: Digital Document Processing, pp. 29–48. Springer, London (2007)
3. Wu, Z., Mitra, P., Giles, C.L.: Table of contents recognition and extraction for heterogeneous book documents. In: Document Analysis and Recognition 12th International Conference, 2, pp. 1205–1209 (2013)
4. Sonka, M., Hlavac, V., Boyle, R.: Image processing, analysis, and machine vision. Cengage Learning (2014)
5. Hu, T.: New Methods for Robust and Efficient Recognition of the Logical Structures in Documents. IIUFUniversité de Fribourg, Switzerland (1994)
6. Satkhozhina, A., et al.: Non-manhattan layout extraction algorithm. In: Proceedings of SPIE-IS&T Electronic Imaging, 86640A (2013)
7. Belaïd, A., D'Andecy, V.P., Hamza, H., Belaïd, Y.: Administrative document analysis and structure. In: Biba, M., Xhafa, F. (eds.) Learning Structure and Schemas from Documents. SCI, vol. 375, pp. 51–71. Springer, Heidelberg (2011)
8. Song, H., Li, L., Zhang, W.: Application of VSM model to document structure identification. Journal of Beijing Information Science and Technology University (Natural Science Edition) **6,** 66–69 (2011)
9. Jin, C.: Determine Algorithm of logical order in document layout based on directed graph. Microcomputer Information **12**, 292–293 (2008)
10. Peng X., Li, N.: Improved VSM algorithm for judging paragraph logic label. Journal of Beijing Information Science and Technology University (Natural Science Edition), 19–24 (2014)
11. Nepomniaschaya, A.S.: An associative version of the bellman-ford algorithm for finding the shortest paths in directed graphs. In: Malyshkin, V.E. (ed.) PaCT 2001. LNCS, vol. 2127, pp. 285–292. Springer, Heidelberg (2001)