Neural Network for Understanding the Structure of Reflowable Documents

**Abstract.** Document structure information plays a key role in the automatic typesetting of document formats. Understanding document structure is necessary to obtain the structure information of documents. Document structure recognition is the key to document understanding.Nevertheless, understanding reflowable document structures, one of the main document formats, is difficult, and current methods for understanding reflowable document structures have nonideal recognition effects. This work proposed a method for recognizing reflowable document structures. The proposed method was based on a neural network and the features of reflowable documents. It exploits the advantages provided by the deep learning of neural network to solve the problem of sequence annotation. The format, content, and semantic features of documents were first extracted by combining the features of reflowable documents obtained by domestic and foreign research. The deep and cross network model was introduced, and feature crossing was applied in an automated manner. Second, the neural network recognition model was constructed by using the long- and short-term memory. The addition of 18 classes improved the recognition accuracy of the model for the logical label and structure of the document. Finally, the transfer method was applied for preliminary report recognition.Experiments showed that the ability of the proposed model to recognize document structure was better than of other machine learning models or methods. Moreover, the proposed model was more effective than current best-performing products.

**Keywords:** Document structure understanding, Deep and cross network, Long- and short-term memory

1. Introduction

Research on reflowable document structure aims to understand the semantic role of document elements in document information extraction and to identify the corresponding relationship between logical document labels and elements. Document structure recognition is the key to understanding documents. It involves recognizing the basic units of paragraphs in each document. These units have different roles, such as titles, abstracts, and tables, in the article. Their corresponding logical labels are then identified through document structure recognition. Document structure recognition is the basis of numerous applications, such as format optimization and so on.

Previous works on document structure recognition have mainly focused on document layout and rarely paid attention to reflowable documents. At present, rule-based or machine learning-based methods are the main methods used for reflowable document structure recognition. However, the recognition effect of these methods is not ideal enough. In addition, rules for the machine method are established in relation to document type. In other words, changing to another kind of document requires re-establishing a new rule or recognition model. This requirement, however, causes the generalization ability of rule-based or machine learning-based methods to deteriorate. In this work, we attempt to use the neural network method to overcome the shortcomings of previous research and to obtain improved recognition results.

1. Related research

Mature methods for the analysis of fixed document layouts exist at home and abroad [1][2]. These methods aim to study the relationship between each field of the layout and logical label and are related to reflowable document structure recognition. Format and content features are extracted from the document to develop the feature representation of the document[3][4][5]. Document structure recognition can be improved by adding semantic features along with formatting features on the basis of textual content. For example, [6] the recurrent neural network (RNN) is used to learn the typesetting format, content, and semantic features of documents. The RNN then expresses these features in n-gram. These methods present good flexibility, fault tolerance, and recognition ability. In addition, some mature products for layout understanding, such as Founder flying software [7], are available. These products represent documents, such as reflowable documents, in the form of XML through layout analysis and then obtain typesetting results that correspond to different templates. However, the main solution to layout analysis is the standardization of the labeling problems of the publication layout area and the recognition of the logical structure of the document is not its main goal. In addition, the structure of reflowable documents is considerably more complex than that of document layouts and often contains typesetting errors. Methods for the structure recognition of reflowable documents are mainly dominated by rule-based and machine learning-based methods.

In the rule-based method [8][9], the logical label of the document to be tested is recognized by comparing the document to be tested with the standard template [10][11][12]. The document structure is then determined on the basis of the document typesetting format and textual content in accordance with predetermined rules. This method must construct numerous rules given the highly complex structure of reflowable documents. In addition, this method has inadequate fault tolerance for poorly typeset documents.The logical label classifier is based on the machine learning method. It is constructed by a learning model, which does not require rules and has good error tolerance. For example, the construction of a document structure recognition model based on SVM and random forest involves the application of eight typesetting features, including font type and size [13]. Content feature has been expanded. CRF has been used to learn document features on the basis of formatting features, and the model for document structure recognition has been constructed [14]. Moreover, most of these methods only apply a small number of format and content features and have low recognition accuracy.

Document structure recognition becomes a sequence annotation problem if each part of a reflowable document is regarded as a basic typesetting unit, and the structure of a reflowable document can be regarded as a sequence that is based on these basic units. However, existing machine learning methods seldom consider the long-distance dependence between document units.

In this work, we attempt to use the deep learning model RNN, which can process and learn effective features from sequential data. Long- and short-term memory (LSTM) is a variant of RNN and has been used effectively to solve the problem of long-term dependence between sequences. LSTM has been widely used in recent years [15]. It has been applied to match the elements of the text sequence with the model input in the task of English annotation. Forward and reverse learning characteristics have been studied by using the bidirectional LSTM neural network structure [16]. Given that machine translation is regarded as a sequence annotation task, LSTM has been used to decode target language sentences [17]. And BiLSTM neural network has been to identify authorship [18].

At the same time, the selection of the features of the document unit is the key to recognizing the logical tag of the document unit. However, judging the specific logical role only from a single feature is insufficient and fails to provide a good recognition effect. Therefore, the interaction between features is particularly important.

Deep neural networks (DNN) can embed vectors and nonlinear activation functions because they can learn nontrivial high-degree feature interactions. The recent success of the residual network [19]has enabled the training of very deep networks. Deep crossing [20] extends residual networks and achieves automatic feature learning by stacking all types of inputs but is not the most efficient method for feature engineering. The wide-and-deep model [21] was established in this spirit. It takes cross-features as inputs into a linear model and jointly trains the linear model with a DNN model. However, the success of the wide-and-deep model hinges on the appropriate selection of cross-features. In this work, we proposed the deep & cross network (DCN) model that enables automatic feature learning. Our DCN model efficiently captures the effective feature interactions of bounded degrees and learns highly nonlinear interactions. It does not require manual feature engineering or exhaustive searching and has low computational cost.

In addition, neural network training requires a massive document corpus as the training data to ensure that the recognition model of various types of documents has a good effect because the final classification tags of different types of documents are different and the corpus of some types of documents is small. Transfer learning is proposed to solve this problem. Most of the current work in the new research field of transfer learning has focused on algorithm theory. Early works on transfer learning in the field of machine learning include “learning how to learn,” an incremental self-reinforcing learning system [22]. They also include “lifelong learning,” which pointed out that the ability to learn knowledge transfer is crucial for the development of lifelong learning given that the generalization of lifelong learning methods is initiated with only a few pieces of training data [23]. The transferability of DNNs in the transfer learning method proposed for deep networks has been extensively investigated [24]. Hu and others applied migration learning to convolutional neural networks and achieved good results [25]. Later, Wang and others proposed a deep migration learning strategy that applies migration learning to deep residual networks to classify images [26].

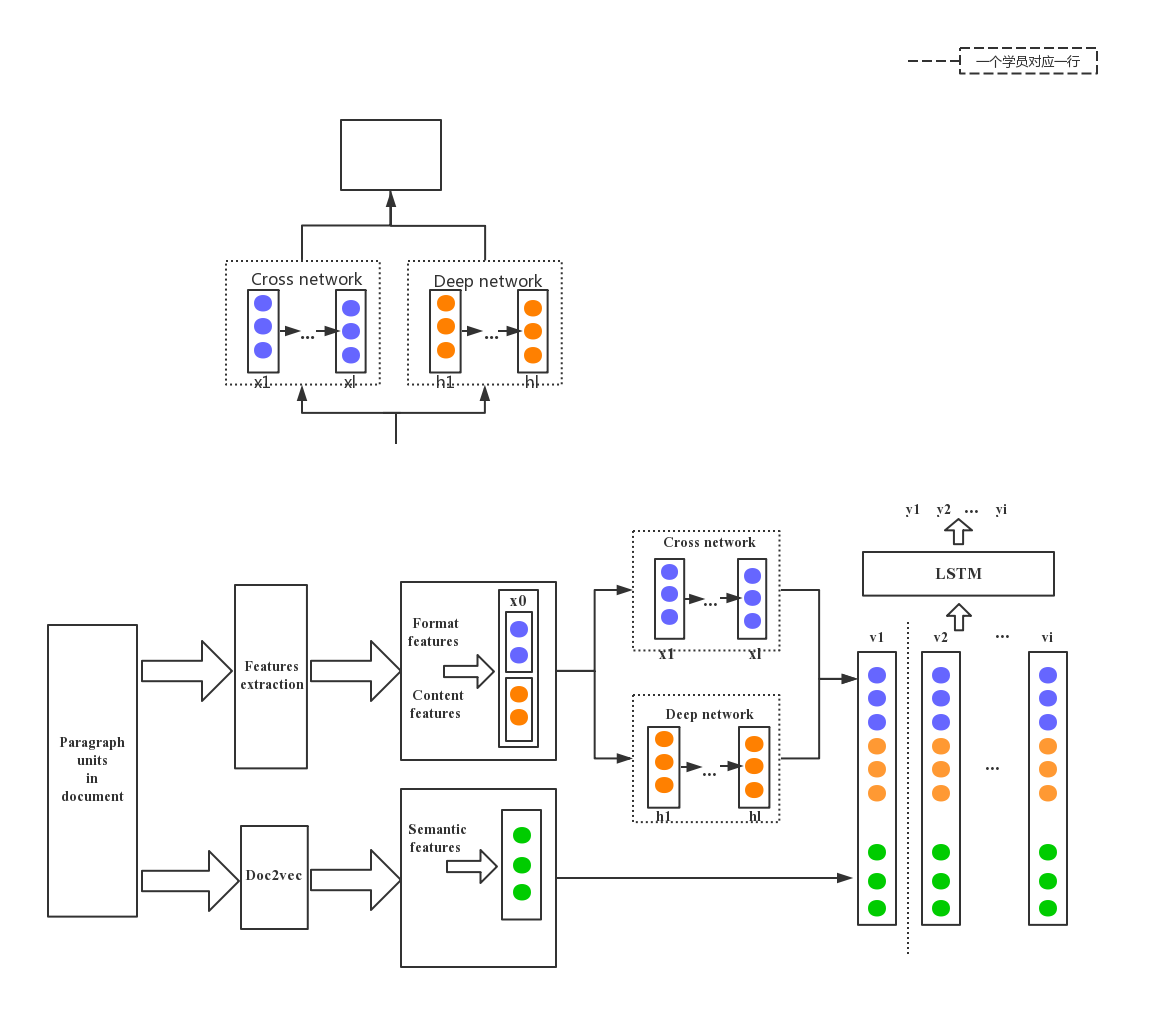
In this work, transfer learning is preliminarily applied. In transfer learning, the weight of each node in a layer of neural network is transferred from a trained network to a new network instead of training a neural network for a specific task from the beginning to optimize the model optimization of different types of documents.

In summary, we attempt to artificially extract three kinds of document features, including content and semantic features, and to construct the DCN model. We then identify interaction features by applying cross-features automatically. Next, we use the LSTM neural network to build the document recognition model for improving recognition ability. Finally, we apply transfer learning technology for the preliminary structure recognition of other types of documents.

1. Model for document structure recognition

The key to document understanding is document structure recognition, which mainly aims to find the correspondence between each unit and the document logical label. In this work, the logical label set is defined as S = {paper title, author name, author unit, Chinese abstract, Chinese keyword, English abstract, English keyword, first-level title, second-level title, third-level title, fourth-level title, text paragraph, figure, figure title, table title, table, formula, program code}.

The framework of document structure recognition includes document feature extraction, deep cross network, and model construction. Typesetting format, document content, and text semantic features are combined as the features of the document unit for feature representation. Our model constructs the cross-feature model on the basis of DCN, that is, feature interactions are automated by combining the cross network with the deep network. The recognition model is constructed on the basis of LSTM, and the global contextual relationship between document units is fully explored to recognize document structure. The recognition model is shown in figure 1.



**Fig. 1.** Document structure model based on the bidirectional LSTM neural network

1. Extraction of document structure features

The selection of document unit features is the key to recognizing logical labels for document units. In this work, relevant features are manually selected, and the model is trained by marked learning samples to improve the efficiency of machine learning. The features extracted in this work include format, content, and semantic features. The format feature reflects differences in typesetting format among document units. The content feature reflects differences in text among current units. The semantic feature reflects differences in semantic content among different units.

1.Format features

1)Font size: the font size of the current paragraph;

2)Font shape: whether the current paragraph is in bold;

3)Outline level: the outline level of the current paragraph;

4)Alignment: the alignment of the current paragraph;

5)Word object: whether the current paragraph is a “graph”, “table”, or “formula”.

Font size, font shape, and outline level have been used in previous methods. By contrast, we are the first to use alignment and word objects. Some document units, figures, and table titles, follow specific rules for center, left, and both-end alignment, whereas pictures, formulas, and table objects have clear object types. Therefore, these features can improve the ability to recognize the relevant document unit.

In addition, we use a high number of efficient relative features to encode several features. Instead, we adopt the concept of relative font size. That is, the title level is determined on the basis of the most frequently used font size, which represents the difference in font size of each document.

2.Content features

1)Number: the number of the first or tail of the document unit;

2)Key words: the existence of specific terms, such as “summary”, and “figure”;

3)Number position: the number is located at the beginning of the paragraph or tail.

Previous works have mainly focused on number and keyword. We find that the number position of some document units also plays an important role. For example, the number position of formulas is usually located at the end of a paragraph.

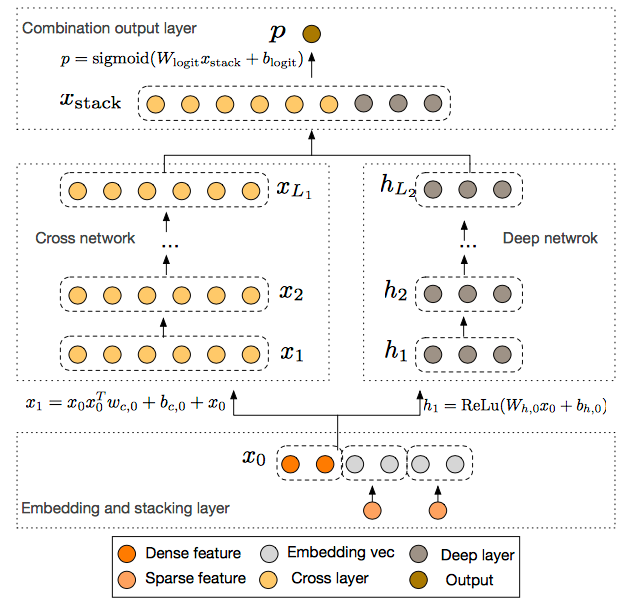
3.Semantic features

Text is the basic content of a document, and its semantic features can also play an auxiliary role in judging the logical structure of a document. A simple learning approach used by neural networks for document structure recognition is to input the text content directly into the network. In this work, we adopt doc2vec to encode paragraph text and represent it as a vector. Thus, the long text in a document can be summarized, and semantic information is preserved.

1. Model of the Deep and Cross network

Through the analysis of the features of logical roles, it can be seen that the document features used by logical roles have strong overlap, and it is not good enough to judge specific logical roles only from a single feature. Therefore, this paper proposes the Deep and Cross network (DCN) model. The input of the DCN model includes the sparse feature, the dense feature, and the embedding feature. The model training phase is divided into two parts. The right part is the deep layer of the traditional DNN model, in which each deep layer is connected to the ReLU active layer. The feature makes the feature more advanced through multiple hidden layers, while the cross layer on the left is in accordance with a recursive feature combination formula. The features of each layer are crossed and combined by the features of the upper layer, and the original features of the upper layer are added back. With the increase of cross layer, it can generate arbitrary high-order cross-combination features. Finally, the output of cross layer and deep layer are combined to train subsequent model. The general model framework is shown in figure 2.





**Fig. 2.** The framework of the Deep and Cross network model

* 1. Cross network

The key idea of the across networks is to apply explicit feature intersections in an efficient manner. The cross network is composed of cross layers which have the following formula:

(1)

where are column vectors denoting the outputs from the and cross layers, respectively; are the weight and bias parameters of the layer. Each cross layer adds back its input after a feature crossing and the map-ping function fits the residual of .

The particular structure of the highly interactive cross network that the cross features have causes the degree of cross features to increase with layer depth. The highest polynomial degree (in terms of input ) for an cross network is . In fact, the cross net-work comprises all the cross terms of degree from 1 to . [27].

For complexity analysis, let denote the number of cross layers, and denote the input dimension. Then, the number of parameters involved in the cross network is

.

* 1. Deep network

There are fewer parameters for the crossover network, which limits the capacity of the model. To capture the interaction between highly nonlinear features, we introduced a deep network parallelism.

The deep network is a fully-connected feed-forward neural net- work, with each deep layer having the following formula:

(2)

where are the and hidden layer, respectively; are parameters for the deep layer; and is the ReLU function.

For complexity analysis, we assume that all deep layers have the same size to simplify. Let denote the number of deep layers and denote the deep layer size. Then, the number of parameters in the deep network is

Final, the combination layer concatenates the outputs from two networks whose result is that represents concatenated vectors of interacting features. And then feed the concatenated vector into the neural network behind.

We jointly train both networks, as this allows each individual network to be aware of the others during the training.

1. Model of the bidirectional LSTM neural network

As mentioned earlier, we consider reflowable document structure recognition as a sequence annotation problem that is based on the document unit. Therefore, we construct a model of structure recognition that is based on the LSTM network.

LSTM redesigns the memory module of the hidden layer node on the basis of the traditional RNN model. It adds input gate and output gate to adjust the information of the input data and the state information of the memory unit. It adds forget gate to clean up useless information in the learning process and to exploit long-distance sequence information. In addition, the hidden nodes of the LSTM are used to discover the correlation between paragraphs.

We adopt three input layer features: format feature, content feature , and semantic feature . We then set their dimensions as. The input of the model can be expressed as , where . At the same time, we set as the output of the hidden layer in the current paragraph and as the output of the previous moment.

The value of the forgetting gate is calculated as

(3)

The value of the input gate and the candidate values for the current state of the memory unit are then calculated as

(4)

(5)

The state value at the current moment is calculated in accordance with the state at the last moment, the current candidate value , the input of the current state, and the value that needs to be forgotten as follows:

(6)

We calculate the output gate value and the output of LSTM in accordance with and and the hidden state as

(7)

 (8)

where parameter is the logistic sigmoid function, and ; ; and are the dimensions of the hidden layer units of LSTM.

After LSTM, the feature of document unit is calculated to obtain the output ,where .

However, standard LSTM processes sequences in chronological order and considers only historical messages while ignoring future information. The addition of future information has been shown to help mark the model task. Therefore, we use bidirectional LSTM to learn the feature information of document elements and process the sequence of document elements from the forward and reverse directions.

Regularization is crucial for improving the generalization ability of the neural network model. We adopt the regularization operation for the feature representation of document paragraphs. That is, the hidden nodes of the neural network that randomly maintain a certain proportion are not working during model training, and the output is set to 0:

(9)

The probability that dimension of , which consists of 0 and 1, takes a value of 0 follows Bernoulli distribution. Given that the document unit is represented as , the document structure sequence can be represented as , and represents the number of paragraphs in each document. We use the forward and reverse two LSM networks to process the document unit sequence from the forward and reverse directions and generate a positive document unit to represent and a reverse document unit to represent to ensure that the sequence of every unit can obtain information from the context. We obtain document element feature through the corresponding summation operation and use as the activation function. We express sequence document elements as follows:

 (10)

 (11)

Finally, we enter into the Softmax layer and calculate the output of each document unit in the sequence, that is, the probability value of each label of each document unit:

(12)

where is the number of the document label category . The function for any vector is defined as

(13)

Formula (13) represents the probability that sample belongs to class . The vector of paragraph in the document is expressed as , and the output corresponding to Softmax is , which can be understood as the probability of unit belonging to the logical label set. Finally, the most probable value is selected as the current document paragraph label as

(14)

1. Experimental analysis
   1. Experimental data and evaluation indicator

An open reflowable document corpus library for document structure recognition does not exist. Thus, we construct a reflowable document corpus that consists of 82,763 document paragraphs from 1,365 academic papers. These documents are retrieved from academic papers and journals, such as the Journal of Food Science and Technology, Journal of Beijing Information Science and Technology University. These documents are randomly divided into training and test sets in accordance with a 4:1 ratio. As shown in the table, the distribution of the number of logical labels for the 18 types of logical labels is unbalanced. For example, 39.72% of the document element labels are all text segments, and simply using the precision rate P, recall rate R, and F value as evaluation indicators is unreasonable. Thus, we adopt macro- and micro-averages to quantify method performances.

* 1. Experimental comparison

We compare the best-performing methods for reflowable document structure recognition with our proposed method. We select random forest with the best performance for the comparison experiment. We select the same features and feature processing methods and set the value of the parameter tree as 100 to retest the methods. CRF, MEMM, and HMM have been used to classify document paragraphs on the basis of extracted format and content features. We also select the CRF with the best performance for comparison.

**Table 1.** Comparison of the recognition effects of different models

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | The model in the paper | | | | KIM | | | | Lei Y | | | | number |
| P | R | F | P | | R | F | P | | R | F |  | |
| Paper title | 0.99 | 0.98 | 0.99 | 0.85 | | 0.86 | 0.85 | 0.99 | | 0.99 | 0.99 | 2286 | |
| Author name | 0.96 | 0.97 | 0.97 | 0.66 | | 0.73 | 0.69 | 0.94 | | 0.97 | 0.95 | 2219 | |
| Author unit | 0.98 | 0.96 | 0.97 | 0.79 | | 0.74 | 0.76 | 0.97 | | 0.97 | 0.97 | 2290 | |
| Email | 0.99 | 0.99 | 0.99 | 0.99 | | 0.95 | 0.97 | 0.99 | | 0.96 | 0.97 | 2290 | |
| Chinese abstract | 0.99 | 0.97 | 0.98 | 0.88 | | 0.91 | 0.89 | 1.00 | | 0.97 | 0.99 | 1321 | |
| Chinese keyword | 0.97 | 0.99 | 0.98 | 0.94 | | 0.90 | 0.92 | 1.00 | | 0.97 | 0.99 | 1307 | |
| English abstract | 0.98 | 0.97 | 0.98 | 0.96 | | 0.93 | 0.94 | 0.94 | | 1.00 | 0.97 | 1057 | |
| English keyword | 0.98 | 0.99 | 0.99 | 0.87 | | 0.94 | 0.91 | 0.95 | | 1.00 | 0.98 | 1006 | |
| First-level title | 0.98 | 0.97 | 0.98 | 0.81 | | 0.71 | 0.76 | 0.92 | | 0.90 | 0.91 | 5443 | |
| Second-level title | 0.96 | 0.94 | 0.95 | 0.73 | | 0.83 | 0.77 | 0.94 | | 0.95 | 0.95 | 6752 | |
| Third-level title | 0.87 | 0.91 | 0.89 | 0.94 | | 0.84 | 0.89 | 0.87 | | 0.91 | 0.89 | 1513 | |
| Text paragraph | 0.94 | 0.91 | 0.92 | 0.86 | | 0.69 | 0.76 | 0.91 | | 0.92 | 0.92 | 32875 | |
| Figure | 0.99 | 0.94 | 0.96 | 0.83 | | 0.84 | 0.83 | 0.95 | | 0.95 | 0.95 | 5158 | |
| Title of figure | 0.98 | 0.98 | 0.98 | 0.90 | | 0.91 | 0.90 | 0.98 | | 0.95 | 0.97 | 6073 | |
| Table title | 0.97 | 0.98 | 0.97 | 0.91 | | 0.79 | 0.85 | 0.89 | | 0.94 | 0.91 | 2252 | |
| Table | 0.98 | 0.97 | 0.97 | 0.92 | | 0.91 | 0.92 | 0.86 | | 0.98 | 0.92 | 2286 | |
| Formula | 0.84 | 0.91 | 0.88 | 0.92 | | 0.90 | 0.91 | 0.96 | | 0.90 | 0.93 | 2219 | |
| Program code | 0.59 | 0.36 | 0.44 | 0.66 | | 0.40 | 0.50 | 0.53 | | 0.36 | 0.42 | 443 | |
| Macro average | 0.94 | 0.93 | 0.93 | 0.86 | | 0.82 | 0.83 | 0.92 | | 0.92 | 0.92 |  | |
| Micro average | 0.96 | 0.96 | 0.96 | 0.85 | | 0.86 | 0.85 | 0.92 | | 0.94 | 0.93 |  | |

Table 1 shows the following:1) Our model for document structure recognition is generally superior to the other two models. There are 14 of the F-values that are the highest of the 18 logical labels. In addition, the F values of the macro- and micro-averages of our model are better than those of other models and are 1% and 3% higher than those of the model of Lei et al., respectively, and 10% and 11% higher than those of the model of Kim et al, respectively. These results indicate that the feature set extracted our method and the bidirectional LSTM model constructed by our method can perform document structure recognition.2) Traditional statistical models, such as random forest, have the best recognition effect for formula and program codes.3) The recognition ability of our model is better than that of random forest, which does not consider contextual information. The F value of our model in the recognition of author name, author unit, first-level title, text paragraph, and figure and table titles is 10% higher than those of other models. In particular, the F value of our model for the recognition of first-level titles is 22% higher than that of other models.

In addition, we compare the structure recognition performance of our model with that of Founder flying software. We randomly select 15 papers for comparison. The results are shown in table 2. The results show that all the indicators of our model are better than those of Founder, and the macro- and micro-average F values of our method are 0.08 higher than those of Founder.

**Table 2.** Comparison of the recognition effect of the model with that of Founder flying software

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Founder flying software | | | The model of this paper | | |
|  | P | R | F | P | R | F |
| Chinese title | 1.00 | 0.60 | 0.75 | 0.96 | 1.00 | 0.98 |
| Author name | 0.88 | 0.58 | 0.70 | 0.92 | 0.92 | 0.92 |
| Author unit | 0.94 | 0.94 | 0.94 | 0.96 | 0.92 | 0.94 |
| Chinese abstract | 0.82 | 1.00 | 0.90 | 1.00 | 1.00 | 1.00 |
| Chinese keyword | 0.82 | 1.00 | 0.90 | 1.00 | 1.00 | 1.00 |
| English abstract | 0.93 | 1.00 | 0.96 | 1.00 | 1.00 | 1.00 |
| English keyword | 0.79 | 1.00 | 0.88 | 1.00 | 1.00 | 1.00 |
| Text | 0.90 | 0.89 | 0.90 | 0.94 | 0.99 | 0.96 |
| Figure | 0.90 | 1.00 | 0.95 | 0.87 | 1.00 | 0.93 |
| Figure title | 1.00 | 0.96 | 0.98 | 0.93 | 1.00 | 0.96 |
| Table title | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Table | 1.00 | 1.00 | 1.00 | 1.00 | 0.97 | 0.98 |
| First-level title | 0.65 | 0.94 | 0.77 | 0.96 | 0.96 | 0.96 |
| Second-level title | 0.88 | 0.85 | 0.86 | 0.96 | 0.95 | 0.96 |
| Third-level title | 1.00 | 0.52 | 0.68 | 1.00 | 0.87 | 0.93 |
| Macro average | 0.91 | 0.88 | 0.89 | 0.97 | 0.97 | 0.97 |
| Micro average | 0.89 | 0.86 | 0.87 | 0.95 | 0.96 | 0.95 |

* 1. Analysis of the influence of features

Table 3 shows the influence of the introduction and absence of DCN on document structure recognition. The introduction of DCN results in a slight increment in the micro average F index and an increment of 1% in the macro average F index. These results indicate that the introduction of DCN is helpful in improving the performance of the document structure recognition model.

**Table 3.** Influence of DCN on recognition results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | micro average | | macro average | |
|  | Introduce DCN | Not Introduce DCN | Introduce DCN | Not introduce DCN |
| P | 0.9567 | 0.9549 | 0.9443 | 0.9445 |
| R | 0.9567 | 0.9549 | 0.9056 | 0.8994 |
| F | 0.9567 | 0.9549 | 0.9209 | 0.9155 |

Table 4 shows the influence of different input features on the effectiveness of document structure recognition. Format and content features play a major role and semantic features play a supplementary role in document structure recognition. The best structure recognition result is obtained by combining the three features.

**Table 4.** Influence of different features on recognition results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Macro average | | | Micro average | | |
| P | R | F | P | R | F |
| Format and content | 0.91 | 0.90 | 0.90 | 0.94 | 0.94 | 0.94 |
| Semantic | 0.87 | 0.85 | 0.86 | 0.89 | 0.90 | 0.89 |
| Format, content and semantic | 0.94 | 0.93 | 0.93 | 0.96 | 0.96 | 0.96 |

* 1. Transfer learning and its analysis

In the current corpus, the labels of each type of document are highly detailed, and the labels of different types of documents are different. Thus, the current neural network model cannot be applied to other types of documents. At the same time, the corpus of some types of documents is insufficient, and neural networks require a large number of document corpus as training data. Thus, model training cannot be improved with a small corpus. The currently constructed network model applies paper type as the corpus data. Thus, this model can only be applied to one type of paper and not to other types of documents, such as reports.

We construct a recognition model for report-type documents. The corpus marked for report-type documents remain limited in the current corpus. We propose a method for transfer learning to overcome this problem. Using the above document recognition model for paper type, and using transfer learning method can realize the structure recognition of report-type documents.

The deep learning model of academic papers is used as the basic model, and its output layer labels and loss functions are modified to meet the requirements of document structure recognition of common paper types. Finally, the network is fine-tuned to obtain the final model, which can complete the recognition task and optimize the recognition result well. The experimental results are as follows:

**Table 5.** Influence of transfer learning on the recognition results for report-type documents

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Not use transfer learning | | | Use transfer learning | | | Total |
|  | P | R | F | P | R | F |  |
| First-level title | 0.96 | 0.96 | 0.96 | 0.96 | 0.98 | 0.97 | 475 |
| Second-level title | 1.00 | 0.48 | 0.65 | 1.00 | 0.52 | 0.69 | 46 |
| Author name | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 8 |
| Formula | 0.75 | 1.00 | 0.85 | 0.86 | 1.00 | 0.92 | 6 |
| Key content | 1.00 | 1.00 | 1.00 | 0.99 | 1.00 | 1.00 | 115 |
| Figure | 1.00 | 0.91 | 0.98 | 1.00 | 0.93 | 0.96 | 67 |
| Figure title | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 5 |
| Text | 0.94 | 0.94 | 0.94 | 0.95 | 0.94 | 0.94 | 2051 |
| Name | 0.75 | 0.56 | 0.63 | 0.90 | 0.82 | 0.86 | 11 |
| Signer or organization | 0.95 | 0.83 | 0.89 | 0.90 | 0.90 | 0.90 | 83 |
| Date of signature | 0.90 | 0.82 | 0.86 | 1.00 | 1.00 | 1.00 | 11 |
| Table | 1.00 | 0.91 | 0.96 | 1.00 | 0.92 | 0.96 | 12 |
| Table title | 0.68 | 0.77 | 0.73 | 0.71 | 0.78 | 0.74 | 361 |
| Title | 0.99 | 1.00 | 0.99 | 0.99 | 1.00 | 1.00 | 115 |
| average/total | 0.91 | 0.91 | 0.91 | 0.92 | 0.92 | 0.92 | 3366 |

The above table shows the recognition results obtained without and with transfer learning. All the methods using transfer learning achieved the best results among the 14 categories requiring recognition through machine learning methods. As can be seen from the table, each category of labels has a certain degree of improvement, and the improvement is pronounced for the category with few numbers. Given the quantity imbalance of document units and insufficient document corpus, learning effective feature representation from the document structure recognition model without transfer learning is difficult and yields poor results. However, the recognition effect is poor for a large number of document units, such as 2,051 paragraphs of text. Finally, the average F for the model of document structure recognition with transfer learning is 0.92 and is higher than the average F of 0.91 for the model without transfer learning.

1. Conclusion

We propose a method for reflowable document structure recognition. First, the format, content, and semantic features of the document unit are extracted. A DCN model is introduced on the basis of extracted features, and then feature intersection is applied in automatic mode. Second, the recognition of document structure is regarded as a sequence annotation problem, and the recognition model is constructed by using the bidirectional LSTM neural network. Experiments show that the features extracted by our model can be used to distinguish among different types of document units effectively, and the DCN model is important for the performance of the final recognition model. The document structure recognition ability of our model is better than that of other models or methods based on machine learning.

Although our method can accurately recognize 18 types of logical tags and document structure, a gap remains among document typesetting format optimization, document understanding, and other application requirements. 1) Most streaming documents are documents in the preparation process and may contain varying degrees of formatting and content errors. Thus, improving fault tolerance in document structure recognition is necessary. 2) Given the variety of reflowable documents, we will conduct additional in-depth research to enable our model to recognize different types of documents and improve the effectiveness and universality of its ability to recognize document structure.

References

1. **Eskenazi S, Gomez-Krämer P, Ogier J M. A comprehensive survey of mostly textual document segmentation algorithms since 2008[J]. Pattern Recognition, 2016, 64:1-14.**
2. **Mao S, Rosenfeld A, Kanungo T. Document structure analysis algorithms: a literature survey[C]. Electronic Imaging 2003. International Society for Optics and Photonics, 2003: 197-207.**
3. **Tao X, Tang Z, Xu C. Contextual modeling for logical labeling of PDF documents ☆[J]. Computers & Electrical Engineering, 2014, 40(4):1363-1375.**
4. **Tao X, Tang Z, Xu C, et al. Logical Labeling of Fixed Layout PDF Documents Using Multiple Contexts[C]// Iapr International Workshop on Document Analysis Systems. IEEE, 2014:360-364.**
5. **Dong Y, Li Z, Ding Y, et al. Constrained Conditional Random Field for Semantic Annotation of Web Data [J]. Computer Research and Development, 2012, 49(2):361-371.**
6. **Rahman M, Finin T. Understanding the Logical and Semantic Structure of Large Documents[J]. 2017.**
7. **http://www.founderfx.cn/product/1012.jhtml**
8. **Chen L, Zeng G, Wang W.** Extraction and logic description for structure trust pattern of information documents**[J]. Application Research of Computer, 2010, 27(12):4624-4629.**
9. **Li J.** **Research on Checking Method of Document Typesetting Based on Template [D]. Beijing:** Beijing Information Science and Technology University**，2012：25-30**
10. **Song H, Li N, Zhang W. Application of VSM model to document structure identification[J].** Journal of Beijing Information Science and Technology University**. 2011(06)**
11. **Peng X. Research on Document Formatting Method Inspection Method Based on Format Index and Graph [D].** Beijing Information Science and Technology University, **2015**
12. **Iorio A D, Peroni S, Poggi F, et al. Recognizing document components in XML-based academic articles[C]// ACM Symposium on Document Engineering. ACM, 2013:181-184.**
13. **Paaß G, Konya I. Machine Learning for Document Structure Recognition[M]// Modeling, Learning, and Processing of Text Technological Data Structures. Springer Berlin Heidelberg, 2011:221-247.**
14. **Lei Y, Tian Y, Li N, et al. Document Structure Identification Method Based on Conditional Random Field[C]// International Conference on Mechatronics, Control and Materials. 2016.**
15. **Sundermeyer M, Ney H. From feedforward to recurrent LSTM neural networks for language modeling[J]. IEEE/ACM Transactions on Audio Speech & Language Processing, 2015, 23(3):517-529.**
16. **Shao Y, Hardmeier C, Tiedemann J, et al. Character-based Joint Segmentation and POS Tagging for Chinese using Bidirectional RNN-CRF[J]. 2017.**
17. **Sutskever, Ilya, Vinyals, Oriol, Le, Quoc V. Sequence to sequence learning with neural networks[J]. 2014, 4:3104-3112.**
18. Xu X, Cai M, Lu T. Basics depth academic learning Chinese fumiohiro writer authorship identification research[J/OL]. Application Research of Computers,2018:1-5.
19. He K, Zhang X, Ren S, and Sun J. 2015. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (2015).
20. Ying S, Ryan H, Jiao J, Wang H, Yu D, and JC Mao. 2016. Deep Crossing: Web-Scale Modeling without Manually Cra ed Combinatorial Features. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 255–262.
21. Heng-Tze C, Levent K, Jeremiah H, Tal S, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, and others. 2016. Wide & Deep Learning for Recommender Systems. arXiv preprint arXiv:1606.07792 (2016).
22. chmidhuber J. On learning how to learn learning strategies. Technical Report FKI-198-94[R]. Fakultatfur Informatik, 1994．
23. Thrun S. Is learning the n-th thing any easier than learning the first? [J]. Advances in Neural Information Processing Systems, 1996:640-646．
24. Yosinski J, Clune J, Bengio Y, et al. How transferable are features in deep neural networks[J]. neural information processing systems, 2014:3320-3328.
25. Hu M, Chen X, Sun Y, Shen X, et al. A Disease Prediction Model based on Dynamic Sampling and Transfer Learning[J]. Chinese Journal of Computers, 2019:1-18.
26. Wang L, Li J, et al. Application of Deep Transfer Learning in Hyperspectral Image Classification[J]. Computer Engineering and Applications, 2019(05):181-186.
27. Wang R, Fu B, Fu G, et al. Deep & Cross Network for Ad Click Predictions[J]. 2017.