Computers, Materials & Continua
DOI: 10.32604/cmc.2022.025711
Article

Tech Science Press

LAME: Layout-Aware Metadata Extraction Approach for Research Articles

Jongyun Choi[1], Hyesoo Kong[2], Hwamook Yoon[2], Heungseon Oh[3] and Yuchul Jung[1,*]

[1]Department of Computer Engineering, Kumoh National Institute of Technology (KIT), Gumi, Korea
[2]Korea Institute of Science and Technology Information (KISTI), Daejeon, Korea
[3]School of Computer Science and Engineering, Korea University of Technology and Education (KOREATECH), Cheonan, Korea
*Corresponding Author: Yuchul Jung. Email: jyc@kumoh.ac.kr
Received: 02 December 2021; Accepted: 27 January 2022

Abstract: The volume of academic literature, such as academic conference papers and journals, has increased rapidly worldwide, and research on meta-data extraction is ongoing. However, high-performing metadata extraction is still challenging due to diverse layout formats according to journal publishers. To accommodate the diversity of the layouts of academic journals, we propose a novel LAyout-aware Metadata Extraction (LAME) framework equipped with the three characteristics (e.g., design of automatic layout analysis, construction of a large meta-data training set, and implementation of metadata extractor). In the framework, we designed an automatic layout analysis using PDFMiner. Based on the layout analysis, a large volume of metadata-separated training data, including the title, abstract, author name, author affiliated organization, and keywords, were automatically extracted. Moreover, we constructed a pre-trained model, Layout-MetaBERT, to extract the metadata from academic journals with varying layout formats. The experimental results with our metadata extractor exhibited robust performance (Macro-F1, 93.27%) in metadata extraction for unseen journals with different layout formats.

Keywords: Automatic layout analysis; layout-MetaBERT; metadata extraction; research article

1 Introduction

With the development of science and technology, the number of related academic papers distributed periodically worldwide has reached more than several hundred thousand. However, their layout styles are as diverse as their subjects and publishers, although the portable document format (PDF) is widely used globally as a standardized text-based document provision format. For example, the information order is inconsistent when converting such a document to text because no layout information separating the document content is provided. Thus, extracting meaningful information such as metadata, including title, author names, affiliations, abstracts, and keywords, from a document is quite challenging.

Research on extracting metadata or document objects from PDF documents using machine learning has increased [1–7]. In aspects of natural language processing (NLP) approach, Open-source software, such as Content ExtRactor and MINEr (CERMINE) [4] and GeneRation of Bibliographic Data (GROBID) [5], automatically extract metadata using the sequential labeling technique but generally do not take the layouts into account in detail. Therefore, they do not show reasonable metadata extraction performances for every research article due to their diverse (and sometimes bizarre) layout formats.

Unlike existing NLP based metadata extraction approaches, PubLayNet [1], LayoutLM [7], and DocBank [3] employ object detection models, such as Mask region-based convolutional neural network (Mask R-CNN) [8] and Faster R-CNN [9], to detect the layout of academic literature and extract document objects (e.g., text areas, figures, tables, titles, lists, and so on). For example, the PubLayNet-based model detects the layouts of PubMed papers well. However, when other documents that do not appear in training data are given, it omits some information fields or fails to extract the correct regions of documents objects, as shown in Fig. 1. In addition, Fig. 1a shows inconsistent extraction results, such as missing author information or detecting a two-column layout as a one-column layout. In the case of Fig. 1b, the model successfully extracts the layout of the English title but fails to detect that of the Korean title. Moreover, it generates strange detections in the middle of the English abstraction layout. To mitigate the symptoms in the layout detection task, we decided to construct a pre-trained model from various documents with various layouts.

In terms of training data and its coverage, PubLayNet and LayoutLM automatically construct the training data using the metadata provided by PubMed Central Open Access-eXtensible Markup Language (PMCOA-XML) or LaTex. Nevertheless, these are primarily for extracting figures and tables; they do not cover all the necessary metadata, such as the abstract, author name, keyword, or other data [1]. Moreover, to the best of our knowledge, the PMCOA-XML data of papers are only limited to biomedical journals, and small numbers of LaTex data are available in the public domain. Recently, some training data for metadata extraction with consideration of the layout for the selected 40 Korean scientific journals were manually crafted [6]. However, its layout-aware data quality is not so satisfactory due to inconsistent and noisy annotations.

To guarantee consistent annotation quality in constructing layout-aware training data and a more sophisticated language model for advanced metadata extraction, we propose a LAyout-aware MEta-data extraction (LAME) framework composed of three key components. First, an automatic layout analysis for metadata is designed with PDF information extracted by python library PDFMiner[1]. Second, a large amount of layout-aware metadata is automatically constructed by analyzing the first page of papers in selected journals. Finally, we implemented a metadata extractor that contains Layout-aware Bidirectional Encoder Representations from Transformers for Metadata (Layout-MetaBERT) models that follow the BERT architecture [10].

In addition, to show the effectiveness of the Layout-MetaBERT models, we performed a set of experiments with other existing pre-trained models and compared them with the state-of-the-art (SOTA) model (i.e., bidirectional gated recurrent units and conditional random field (Bi-GRU-CRF)) for metadata extraction.

Our main contributions are as follows:

- We proposed an automatic layout analysis method that doesn't require PMCOA-XML (or Latex) data for metadata extraction.

[1] https://github.com/pdfminer/pdfminer.six

- We automatically generated training data for the layout-aware metadata from 70 research journals (65,007 PDF documents).
- We constructed a new pre-trained language model, Layout-MetaBERT, to deal with the metadata of research articles having varying layouts.
- We demonstrated the effectiveness of Layout-MetaBERT on ten unseen research journals (13,300 PDF documents) with diverse layouts compared with the existing SOTA model (i.e., Bi-GRU-CRF).

Figure 1: Layout analysis results of PubLayNet [1]

2 Related Work

2.1 Metadata Extraction

Various attempts have been made to analyze and extract information from documents and classify them into specific categories. Studies on text classification have been continuous since 1990, and the performance of text classification has gradually improved with the employment of sophisticated machine learning algorithms, such as the support vector machine (SVM) [11], conditional random fields (CRF) [12], convolutional neural network (CNN) [13], and bidirectional long short-term memory (BiLSTM) [14]. Afterward, various successful cases using Bidirectional Encoder Representations

from Transformers (BERT) [10] pre-trained with a large-scale corpus were introduced in the field of NLP. In the studies by [15,16], the pre-trained BERT model was fine-tuned on the text classification task, and it showed results close to or superior to the SOTA result for the target data. BERT-based pre-trained models became popular due to their high performances in various NLP fields, and more advanced pre-trained models [17–19] were introduced according to various research purposes.

As a previous SOTA model of our metadata extraction task, a Bi-GRU-CRF model trained more than 20,000 human-annotated pages of layout boxes for metadata [6] from research articles showed an 82.46% of F1-score. However, accurately detecting and extracting regions for each type of metadata in documents is still a nontrivial task because of the various layout formats.

2.2 Document Layout Analysis

Document layout analysis (DLA) [7] and several PDF handling efforts [6,11,20] have been conducted to understand the structure of documents. The DLA aims to identify the layout of text and nontext objects on the page and detect the layout function and format. Recently, the LayoutLM model [7] employed three different information elements for BERT pre-training to identify layouts: 1) layout coordinates, 2) text extracted using optical character recognition software, and 3) image embedding by understanding the layout structure through image processing. Moreover, NLP-based DLA research on various web documents [21], Layout detections and layout creation methods to find text information and location [8,22,23] have been studied. In [2,24] applied the object detection technique to text region detection. Interestingly, widely used object detection techniques (e.g., Mask R-CNN [8] and Faster R-CNN [9]) have been applied to the metadata extraction field [1,3].

Due to the high cost of training data construction for DLA, many studies have attempted to build datasets automatically. For example, the PubMed Central website, which includes academic documents in the biomedical field, provides a PMCOA-XML file for each document, enabling an analysis of the document structure. In the case of PubLayNet [1], which utilizes the PubMed dataset, the XML and PDFMiner's TextBoxes were matched to construct about 1 million training data. However, this is generally possible only when accurate coordinates are provided to separate each layout and the text information elements for each.

3 Proposed Framework

Fig. 2 depicts our LAME framework consisting of three major components: automatic layout analysis, layout-aware training data construction, and metadata extractor. Stage 1 analyzes the given PDF's first-page layout by using PDFMiner. However, due to the incompleteness of the parsing results of the PDFMiner, it undergoes a set of reconstruction, refinement, and adjustment procedures for identifying several metadata that appeared on the first page. In stage 2, we build many training data used in stage 3. The building process matches the identified metadata in stage 1 with the previously existing correct metadata values. Finally, we implement a novel metadata extractor by pre-training our Layout-MetaBERT model with the training data of stage 2 and fine-tuning it for target corpus.

3.1 Automatic Layout Analysis

To understand the layout that separates each metadata element in the given PDF file, we must observe the text and coordinate information on the document's first page. To this end, we employ the open-source software, PDFMiner, to extract meaningful information surrounding the text in the PDF files. For example, if we parse a PDF document with the software, we obtain information on the page, TextBox, TextLine, and character (Char) hierarchically, as illustrated in Fig. 3. These include

various text information, such as coordinates, text, font size, and font for each object. For example, text coordinates appear in the form of (x, y) coordinates along with the height and width of the page.

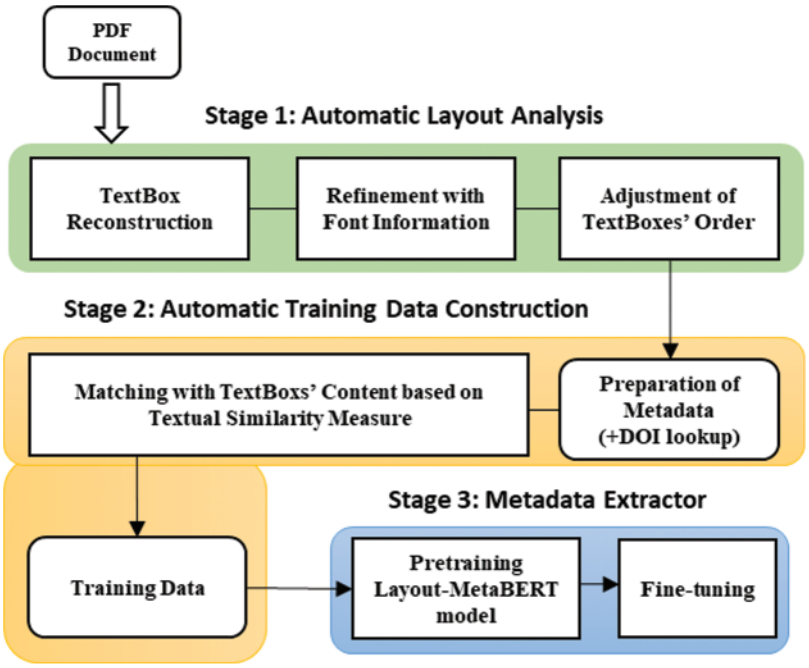


Figure 2: Layout-Aware Metadata Extraction (LAME) framework

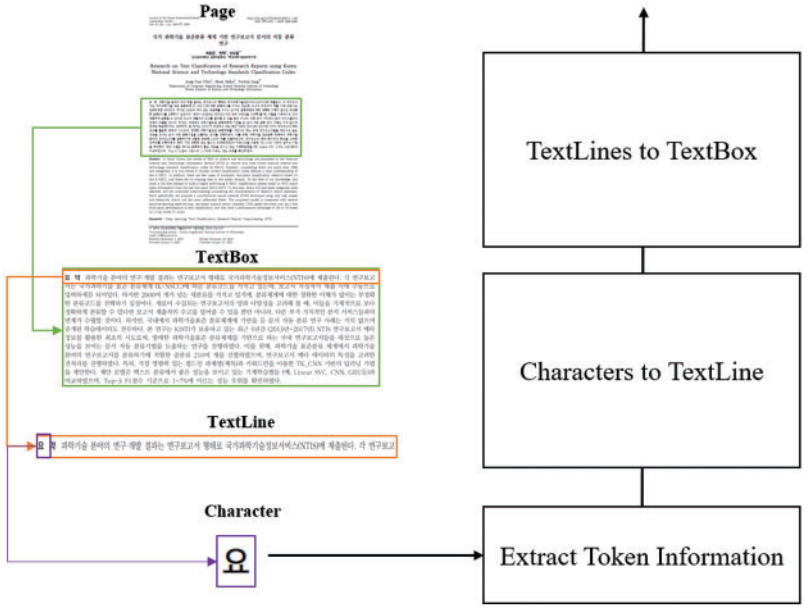


Figure 3: TextBox reconstruction based on the results of PDFMiner

3.1.1 Textbox Reconstruction

It is possible to extract document layout by utilizing PDFminer, but this conveys erroneous information, which is hard to use directly. Therefore, to reduce existing errors in TextBox recognition, as depicted in Fig. 3, TextBoxes were reconstructed starting from the Char unit with the information obtained from the PDFMiner. The detailed steps are as follows:

1) Extract Token Information: First, we extract all characters that appear on the first page of the PDF document and their information.
2) Characters to TextLine: Second, the spacing between characters is analyzed using the coordinate information for each Char. Generally, each token's x-coordinate distance (i.e., character spacing) appears the same, but the distance is slightly different depending on alignment methods or language. Therefore, after collecting characters in the same y coordinate, the corresponding characters were sorted based on the x-coordinate value. Sometimes academic papers have two columns, so different lines may exist in the same Y coordinate. If the distance between two characters is smaller than the font size, we regarded them as one line.
3) TextLines to TextBox: Finally, after aligning the TextLines based on the y-axis, if the distance between each y-coordinate is smaller than the height of each TextLine, the two different TextLines are regarded as the same TextBox. However, this method cannot create a TextBox accurately by separating paragraphs from paragraphs. For more elaborate TextBox composition, it needs to decide whether to configure the TextBox by considering the left x-coordinate x_0, the right x-coordinate x_1, and the width (W) of each TextLine. For example, for sentences like those in Fig. 4, we can think of two cases composing a TextBox by comparing each TextLine. First, the beginning of a paragraph is usually indented. Therefore, if the difference between the x_0 values of L_i and L_{i-1} is greater than the font size of the Chars existing in each TextLine, the two TextLines should be included in different TextBoxes. Second, a TextLine that appears at the end of a paragraph has a shorter width because it has fewer Chars on average. Therefore, when the width of L_{i-1} is smaller than the width of L_i, L_{i-1} and L_i should be assigned different TextBoxes.

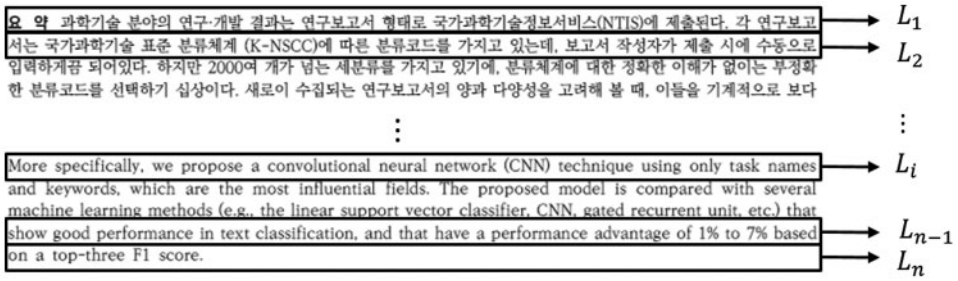


Figure 4: Example of TextBox separation

3.1.2 Refinement with Font Information

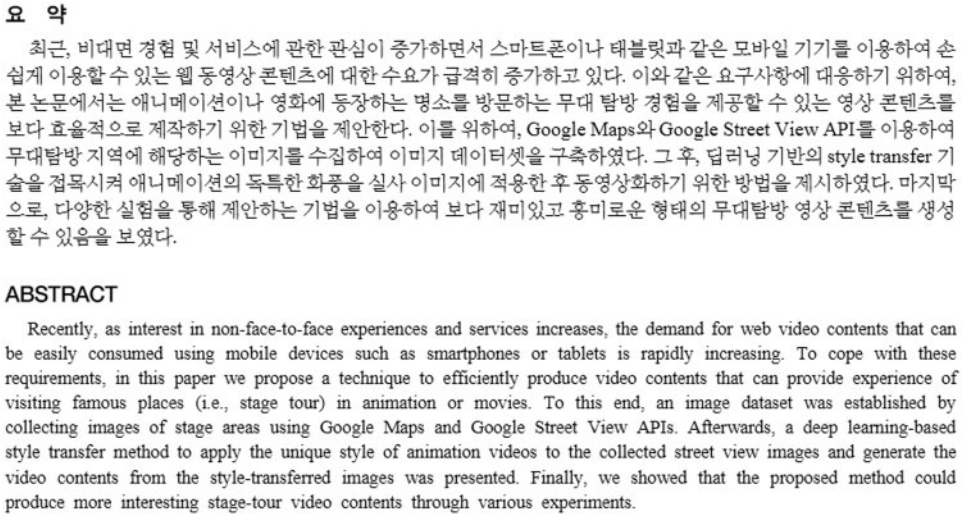
PDFMiner can produce various pieces of information in terms of font information, such as the font name and style (e.g., bold, italic, etc.) as listed in Tab. 1.

However, as in Fig. 5, English is frequently used for Korean abstracts in some journals published in Korea. In particular, abstracts written in Korean and English appear together on the first page of some research articles. In addition, certain strings are often treated as bold or italic and often have

different fonts and sizes, such as section titles. Considering this problem, when composing a TextBox using the coordinate information described above, if the font information displayed on each line is different, it is not simply judged as a different line. After analyzing the font information of different languages that appear, the TextBox was determined by considering the number of the appearing fonts (e.g., bold and italic).

Table 1: Example of font information when PDFMiner is applied

Text	PDFMiner output font name
Abstract	ELNFKM+KoreanGD-Bold-KSCpc-EUC-H
Abstract	INPILL+Gulim
Abstract	Arial-BoldItalicMT
ABSTRACT	GPJCIE+YDIYGO130



요 약

최근, 비대면 경험 및 서비스에 관한 관심이 증가하면서 스마트폰이나 태블릿과 같은 모바일 기기를 이용하여 손쉽게 이용할 수 있는 웹 동영상 콘텐츠에 대한 수요가 급격히 증가하고 있다. 이와 같은 요구사항에 대응하기 위하여, 본 논문에서는 애니메이션이나 영화에 등장하는 명소를 방문하는 무대 탐방 경험을 제공할 수 있는 영상 콘텐츠를 보다 효율적으로 제작하기 위한 기법을 제안한다. 이를 위하여, Google Maps와 Google Street View API를 이용하여 무대탐방 지역에 해당하는 이미지를 수집하여 이미지 데이터셋을 구축하였다. 그 후, 딥러닝 기반의 style transfer 기술을 접목시켜 애니메이션의 독특한 화풍을 실사 이미지에 적용한 후 동영상화하기 위한 방법을 제시하였다. 마지막으로, 다양한 실험을 통해 제안하는 기법을 이용하여 보다 재미있고 흥미로운 형태의 무대탐방 영상 콘텐츠를 생성할 수 있음을 보였다.

ABSTRACT

Recently, as interest in non-face-to-face experiences and services increases, the demand for web video contents that can be easily consumed using mobile devices such as smartphones or tablets is rapidly increasing. To cope with these requirements, in this paper we propose a technique to efficiently produce video contents that can provide experience of visiting famous places (i.e., stage tour) in animation or movies. To this end, an image dataset was established by collecting images of stage areas using Google Maps and Google Street View APIs. Afterwards, a deep learning-based style transfer method to apply the unique style of animation videos to the collected street view images and generate the video contents from the style-transferred images was presented. Finally, we showed that the proposed method could produce more interesting stage-tour video contents through various experiments.

Figure 5: Example of when a Korean abstract and an English abstract exist together

Although font information helps the layout composition, it is still confusing when the same font information is used for individual information marking or bold processing for emphasis or different metadata. Additional processing is required to correctly connect individual fonts to make a layout using the font information. Therefore, we compared only texts described in Korean and English and used only the fonts of the same language to determine the layout.

3.1.3 Adjustment of Text Box Order

Academic papers may consist of one or two columns depending on the format for each journal. In some cases, only the main body consists of two columns, and the title, abstract, and author name are displayed in one column. For example, in Fig. 3, such information as the title and author name was arranged in the center, but the document object identifier (DOI) information or academic journal names appeared separately on the left and right sides. To effectively identify metadata consistently from varied layout formats, we sorted the textboxes extracted from the first pages of the research articles sequentially from top to bottom based on the y-axis.

3.2 Automatic Training Data Construction

We compared the content from the layout text that extracted stage.1 with the metadata prepared in advance to construct the layout-aware metadata automatically. If no metadata is available for the given research article, metadata can be automatically obtained through the DOI lookup. Therefore, this technique can be extended to all journal types where the registered DOI exists.

However, the compared textual content is not always precisely matched. Therefore, to determine the extent of the match, we allowed only fields with almost identical (or high similarity) matches for each layout text information element automatically acquired in the previous step as training data. We used a mixed textual-similarity measure for efficient computation based on the Levenshtein distance and bilingual evaluation understudy (BLEU) score.

The Levenshtein distance was calculated using Python's fuzzywuzzy[2]. The scores calculated using the BLEU [25] measure were summed to determine whether the given metadata displays a degree of agreement of 80% or more. Nevertheless, some post-processing is required in the process. In analyzing the text after extraction, some problems occur when dealing with expression substitutions (e.g., "<TEX>," cid:0000). Encoding errors reduced the portion of mathematical expressions that can be removed as much as possible, and we excluded the text with encoding problems to avoid these errors.

3.3 Metadata Extractor

To implement our metadata extractor, we newly pre-trained a layout-aware language model, so-called, Layout-MetaBERT, that can effectively deal with metadata from research articles. Although pre-training a BERT model requires a large corpus and a long training time, a fine-tuning step can make a difference in performance depending on the characteristics of the data used for pre-training. For example, when pre-trained with specific domain data, such as SciBERT [19] and BioBERT [26], they performed better than Google's BERT model [10] in downstream tasks of science and technology or medical fields. However, to our best knowledge, there is no pre-trained model designed to extract metadata based on research article data.

Fig. 6 describes how the previously constructed training data are used for pre-training and fine-tuning the Layout-MetaBERT. The pre-training stage is to construct a general-purpose layout-aware metadata language model. Meanwhile, the fine-tuning stage aims to build an optimized metadata classification model for targeting corpus. Unlike the Google BERT model [10], in our Layout-MetaBERT pre-training, each document layout was considered a sequence. Thus, each layout was classified by the [SEP] token to prepare the training data, but other pre-training procedures and hyperparameters are the same as BERT [10]. At this time, pre-training loss is the sum of Next Sentence Prediction (NSP) loss and Masked Language Modeling (LM) loss. The Masked LM Loss computes

[2]https://github.com/seatgeek/fuzzywuzzy

the cross-entropy loss value of the predicted token y_{ij} for the masked token $\hat{y}_{ij}$:

$$L_{MLM} = -\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{C_{vocabsize}} y_{ij} log(\hat{y}_{ij}) \tag{1}$$

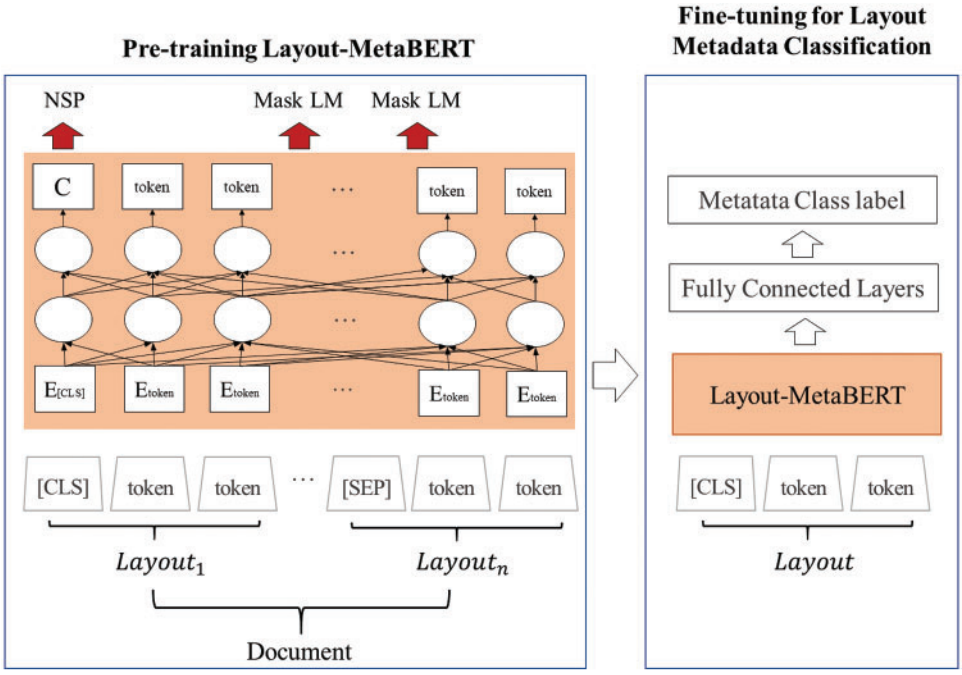


Figure 6: Pretraining layout-MetaBERT and fine-tuning for classification downstream task

Previously, the NSP was used to predict whether two input sentences are consecutive sentences or not. However, in this work, the NSP predicts whether two inputted layouts are consecutive or not. The NSP loss computes the binary cross-entropy loss value of the predicted $\hat{y}_i$ for the y value indicating whether it is a pair or not.

$$L_{NSP} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i) \tag{2}$$

Final loss is:

$$L = L_{MLM} + L_{NSP} \tag{3}$$

In pretraining Layout-MetaBERT models, we followed three size models of the Google BERT: base (L = 12, H = 768, A = 12), small (L = 4, H = 512, A = 8), and tiny (L = 2, H = 128, A = 2), where L is the number transformer blocks, H is the hidden size, and A is the number of self-attention heads. We used a dictionary of 10,000 words built through the WordPiece [27] mechanism. We automatically generated training data extracted from the first page of 60 research journals among the 70 journals for the pre-training.

In the Fine-tuning step, each Layout text information is used as input. For the classification task, we feed the final LayoutMeta-BERT vector for the [CLS] token into a Fully connected layer, and it

computed softmax we can obtain meta classification result. The prediction result loss computes the cross-entropy, and the prediction accuracy computes Macro and Micro F1-score.

4 Experiments

We summarize the results of three major components to examine the applicability of the LAME framework. First, we compare the proposed automatic layout analysis results with other layout analysis techniques. Second, we describe the statistics of the training data constructed according to the results of the automatic layout analysis. Finally, we compare metadata extraction performances of our constructed Layout-MetaBERT models with other deep learning and machine learning techniques after fine-tuning.

4.1 Comparison with Other Layout Analysis Methods

No correct answers exist for the target research articles; thus, we compared the generated layout boxes from PDFMiner, PubLayNet, and the proposed layout analysis method for two randomly selected documents (e.g., A and B) as depicted in Fig. 7.

In Fig. 7, when PDFMiner performs a layout analysis, it generates multiple boxes for abstract and each section (A) and extracts empty layouts between sections as errors (B). Sometimes, it cannot separate keywords from the abstract area (B). Therefore, it is challenging to build training data if we use PDFMiner directly without further modifications.

In the case of Detectron2 based model, which is trained with PubLayNet data, it extracts document objects using object detection technique. So, building training data is impossible because text information cannot be obtained if optical character recognition software is not provided. Although it seems almost similar to ours, it easily fails to detect some metadata elements (A) because PubLayNet does not have metadata fields (e.g., author, affiliation, keywords) as training data. They only cover PubMed papers with almost no citations similar layout formats. Thus, when other documents that do not appear in training data are given, it omits some information fields or fails to extract the correct regions of documents objects (B), and shows inconsistent extraction results, such as missing author information or detecting a two-column layout as a one-column layout (B).

The proposed method could generate a good enough layout analysis for the first page of the research articles through the comparisons. Comparing all three layout analysis results manually for each layout box to calculate the accuracy requires too much human labor and is beyond the scope of this paper. The performance of the constructed Layout-MetaBERT indirectly measures the quality of the layout analysis.

4.2 Training Data Construction

To reflect various kinds of layout formats, we used 70 research journals (Appendix 1) provided by the Korea Institute of Science and Technology Information (KISTI) to extract major metadata elements, such as titles, author names, author affiliations, keywords, and abstracts in Korean and English based on the automatic layout analysis in Section 3.1. Among the 70 journals, two journals were written in only Korean, 23 journals in only English, and 45 in Korean and English.

For each layout that separates metadata on the first page of the 70 journals (65,007 PDF documents), automatic labeling with ten labels was performed, and other layouts not included in the relevant information were labeled O. The statistics of automatically generated training data are presented in Tab. 2.

Figure 7: Layout analysis comparisons with PDFMiner and PubLayNet

4.3 Experimental Results

To check the performance of the proposed Layout-MetaBERT, 70 research journals (65,007 documents) were divided into 60 (51,676 documents) for pre-training (and fine-tuning) and 10 (13,331 documents) for testing, respectively. Tab. 3 lists the training and testing performances of the three Layout-MetaBERT models with widely used metadata extraction techniques. Finally, Tab. 4 describes the Macro-F1 and Micro-F1 scores for metadata classification comparisons with existing pre-trained models.

4.3.1 Fine-tuning and Hyperparameters

In fine-tuning with various pre-trained language models (e.g., three different sized models of Layout-MetaBERT, KoALBERT, KoELECTRA, and KoBERT), all experiments were conducted under the same configurations with an epoch of 5, batch size of 32, learning rate of 2e-5, and maximum

sequence length of 256. In addition, we used the Nvidia RTX Titan 4-way system and Google's TensorFlow framework in Python 3.6.9 for pre-training and fine-tuning.

Table 2: Statistics for automatically generated training data

Metadata field	Label (i.e., layout)	Count
Out of boundary	O	637,856
Title (in Korean)	title_ko	46,056
Title (in English)	title_en	64,414
Affiliation (in Korean)	org_ko	39,233
Affiliation (in English)	org_en	63,434
Abstract (in Korean)	abstract_ko	31,885
Abstract (in English)	abstract_en	55,318
Keywords (in Korean)	keywords_ko	21,685
Keywords (in English)	keywords_en	61,221
Author name (in Korean)	author_name_ko	56,306
Author name (in English)	author_name_en	35,631

Table 3: Train and test performances of metadata extraction

Models	Micro-F1 (train)	Micro-F1 (test)
Layout-MetaBERT (base)	0.9559	0.936
Layout-MetaBERT (small)	0.9595	0.9333
Layout-MetaBERT (tiny)	0.9432	0.9293
KoBERT[3]	0.8086	0.7901
KoalBERT[4]	0.9014	0.8978
KoELECTRA[5]	0.9354	0.9204
Bi-GRU-CRF [6] (without position)	0.8610	0.8912
Bi-GRU-CRF [6] (with position)	0.9442	0.0985
CNN [13]	0.9425	0.824
SVM [11]	0.9411	0.8114

4.3.2 Stable Performances of Layout-MetaBERT

The proposed Layout-MetaBERT models can effectively extract metadata, as listed in Tab. 3. In particular, Layout-MetaBERT models make significant differences compared to the existing SOTA (i.e., Bi-GRU-CRF) model.

Even the tiny model with the fewest parameters among the Layout-MetaBERT models has higher performance than other pre-trained models in Macro-F1 and Micro-F1 scores, as displayed in Tab. 4.

[3] https://github.com/SKTBrain/KoBERT
[4] https://huggingface.co/kykim/albert-kor-base
[5] https://github.com/monologg/KoELECTRA

Moreover, three Layout-MetaBERT models have only minor differences between the Micro-F1 and Macro-F1 scores compared to other pre-trained models. Moreover, the Layout-MetaBERT models exhibit 90% or more robustness in metadata extraction, confirming that pre-training the layout units with the BERT schemes is feasible in the metadata extraction task.

Table 4: Metadata extraction performances of primary BERTmodels for each label

	Layout-MetaBERT (tiny)	Layout-MetaBERT (small)	Layout-MetaBERT (base)	KoBERT	KoALBERT	KoELECTRA
Model size	5 M	16 M	110 M	110 M	12 M	110 M
O	0.94	0.94	0.95	0.85	0.92	0.94
title_ko	0.92	0.94	0.93	0.59	0.89	0.91
title_en	0.92	0.91	0.92	0.76	0.8	0.87
org_ko	0.96	0.96	0.96	0.36	0.96	0.94
org_en	0.86	0.87	0.9	0.64	0.79	0.9
abstract_ko	0.92	0.93	0.93	0.84	0.9	0.92
abstract_en	0.92	0.93	0.94	0.84	0.91	0.91
keywords_ko	0.94	0.95	0.95	0.86	0.94	0.95
keywords_en	0.87	0.89	0.9	0.51	0.91	0.73
author_name_ko	0.97	0.97	0.96	0.75	0.95	0.92
author_name_en	0.56	0.62	0.92	0.3	0.41	0.57
Micro f1	0.9293	0.9333	0.9360	0.7901	0.8978	0.9204
Macro f1	0.8891	0.9009	0.9327	0.6636	0.8527	0.8691

4.3.3 Experiments with Position Information

Unlike other models, the Bi-GRU-CRF model used the absolute coordinates of metadata with other textual features. However, the model failed to discriminate unseen layouts from unseen journals when using the coordinate information for training various journal layout formats. Therefore, to determine the validity of the coordinate information, we performed additional experiments with the Bi-GRU-CRF (with position) and Bi-GRU-CRF (without position) models. Although Bi-GRU-CRF (with position) model demonstrated high performance in the training stage, it failed to recognize metadata-related layouts in unseen journals (less than 10% as F1 score). However, the performance of the Bi-GRU-CRF (without position) model had somewhat lower performance in the training stage compared to the other models. The model performed well, similar to that of KoALBERT. Thus, we confirmed that using absolute coordinate information can only be applied under the premise that the journals used in training also are used in testing.

5 Discussion

5.1 Additional Performance Improvements

The proposed Layout-MetaBERT exhibited higher results than the existing SOTA model [6]. However, absolute coordinate information could obtain poor results for documents in a format not learned. In addition, the proposed layout analysis method separates the metadata well from the first page of the academic documents of various layouts.

However, the accuracy of the automatically generated training data is not perfect. There may be errors due to the difference between the metadata format of the document and the metadata written in advance. As mentioned, encoding errors also occur in extracting text from mathematical formulas or PDF documents. Generating the correct layout significantly affects extracting metadata and is an essential factor in automatically generating data. Therefore, if more sophisticated training data can be generated, the performance of Layout-MetaBERT can be further improved.

5.2 Restrictions of Layout-MetaBERT

Much research has been conducted on automatically extracting layouts from PDF documents. Creating accurate layouts has a significant influence on meta-extraction. This study attempted to compose the layout of the first page of an academic document using text information. Based on this, we trained the Layout-MetaBERT and confirmed the positive results for the applicability to the meta classification module. However, the proposed technique cannot be applied to all documents. For example, an image-type PDF cannot be used unless the text is extracted. In this case, the extraction must be performed using a high-performance optical character recognition module.

5.3 Expansion to Other Metadata Types

This study focused on extracting five major metadata elements (i.e., titles, abstracts, keywords, author names, and author information). Considering that the target research articles contain elements written in English, Korean, or both, the number of metadata becomes 10. However, other metadata (e.g., publication year, start page, end page, DOI, volume number, journal title, etc.) can be extracted further by applying highly refined regular expressions in the post-processing step.

6 Conclusion

This paper proposes the LAME framework to extract metadata from PDFs of research articles with high performance. First, the automatic layout analysis detects the layout regions where metadata exists regardless of the journal formats based on text features, text coordinates, and font information. Second, by constructing automatic training data, we built high-quality metadata-separated training data for 70 journals (65,007 documents). In addition, our fine-tuned Layout-MetaBERT (base) demonstrated excellent metadata extraction performance (F1 = 94.6%) for even unseen journals with diverse layouts. Moreover, Layout-MetaBERT (tiny) with the fewest parameters exhibited superior performance than other pre-training models, implying that well-separated layouts induce effective metadata extraction when they meet appropriate language models.

In future work, we plan to conduct experiments to determine whether the proposed model applies to the more than 500 other journals not used in this study. Moreover, resolving potential errors in the automatically generated training data is a concern to create layouts that separate each metadata element in an advanced way. Furthermore, extending the number of metadata items extracted without post-processing is an exciting but challenging task to resolve as future work.

Funding Statement: This work was supported by the Korea Institute of Science and Technology Information (KISTI) through Construction on Science & Technology Content Curation Program (K-20-L01-C01), the National Research Foundation of Korea (NRF) under a grant funded by the Korean Government (MSIT) (No. NRF-2018R1C1B5031408). In addition, this research is the result of a study on the HPC Support project supported by the Ministry of Science and ICT and NIPA.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] X. Zhong, J. Tang and A. J. Yepes, "Publaynet: Largest dataset ever for document layout analysis," in *Int. Conf. on Document Analysis and Recognition (ICDAR)*, Sydney, NSW, Australia, IEEE, pp. 1015–1022, 2019.

[2] L. Melinda, R. Ghanapuram and C. Bhagvati, "Document layout analysis using multigaussian fitting," in *14th IAPR Int. Conf. on Document Analysis and Recognition (ICDAR)*, Kyoto, Japan, vol. 1, pp. 747–752, 2017.

[3] L. Minghao, X. Yiheng, C. Lei, H. Shaohan, W. Furu *et al.,* "DocBank: A benchmark dataset for document layout analysis," in *28th Int. Conf. on Computational Linguistics*, Barcelona, Spain, pp. 949–960, 2020.

[4] D. Tkaczyk, P. Szostek, M. Fedoryszak and P. Jan, "CERMINE: Automatic extraction of structured metadata from scientific literature," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 18, no. 4, pp. 317–335, 2015.

[5] P. Lopez, "GROBID: Combining automatic bibliographic data recognition and term extraction for scholarship publications," in *Int. Conf. on Theory and Practice of Digital Libraries*, Berlin, Heidelberg, Springer, pp. 473–474, 2009.

[6] S. Kim, S. Ji, H. Jeong, H. Yoon and S. Choi, "Metadata extraction based on deep learning from academic paper in pdf," *Journal of KIISE*, vol. 46, no. 7, pp. 644–652, 2019.

[7] X. Yiheng, L. Minghao, C. Lei, H. Shaohan, W. Furu *et al.,* "Layoutlm: Pre-training of text and layout for document image understanding," in *Proc. 26th ACM SIGKDD Int. Conf. on Knowledge Discovery & Data Mining*, New York, NY, USA, pp. 1192–1200, 2020.

[8] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask r-cnn," in *Proc. IEEE Int. Conf. on Computer Vision*, Venice, Italy, pp. 2961–2969, 2017.

[9] R. Girshick, "Fast r-cnn," in *Proc. IEEE Int. Conf. on Computer Vision*, Santiago, Chile, pp. 1440–1448, 2015.

[10] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, USA, vol. 1, pp. 4171–4186, 2019.

[11] H. Han, C. L. Giles, E. Manavoglu, H. Zha and E. A. Fox, "Automatic document metadata extraction using support vector machines," in *Joint Conf. on Digital Libraries*, Houston, TX, USA, pp. 37–48, 2003.

[12] M. Abramson, "Sequence classification with neural conditional random fields," in *IEEE 14th Int. Conf. on Machine Learning and Applications (ICMLA)*, Miami, Florida, USA, pp. 799–804, 2015.

[13] Y. Kim, "Convolutional neural networks for sentence classification," MS thesis, University of Waterloo, 2015.

[14] P. Zhou, Z. Qi, S. Zheng, J. Xu, H. Bao *et al.,* "Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling," in *Proc. COLING 2016, the 26th Int. Conf. on Computational Linguistics: Technical Papers*, Osaka, Japan, pp. 3485–3495, 2016.

[15] A. Adhikari, A. Ram, R. Tang and J. Lin, "DocBERT: BERT for document classification," arXiv preprint arXiv:1904.08398, 2019. [Online]. Available: http://arxiv.org/abs/1904.08398.

[16] S. Yu, J. Su and D. Luo, "Improving bert-based text classification with auxiliary sentence and domain knowledge," *IEEE Access*, vol. 7, pp. 176600–176612, 2019.
[17] X. Gu, K. M. Yoo and J. Ha, "DialogBERT: Discourse-aware response generation via learning to recover and rank utterances," arXiv preprint arXiv:2012.01775v1, 2021. [Online]. Available: https://arxiv.org/abs/2012.01775.
[18] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma *et al.,* "ALBERT: A lite BERT for self-supervised learning of language representations," in *Int. Conf. on Learning Representations*, New Orleans, USA, 2019.
[19] I. Beltagy, K. Lo and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proc. 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, pp. 3615–3620, 2019.
[20] A. R. Katti, C. Reisswig, C. Guder, S. Brarda, S. Bickel *et al.,* "Chargrid: Towards understanding 2d documents," in *Proc. 2018 Conf. on Empirical Methods in Natural Language Processing*, Brussels, Belgium, pp. 4459–4469, 2018.
[21] Ł Garncarek, R. Powalski and T. Stanisławek, "LAMBERT: Layout-aware (language) modeling for information extraction," in arXiv preprint arXiv:2002.08087, 2020. [Online]. Available: https://arxiv.org/abs/2002.08087.
[22] Z. -Q. Zhao, P. Zheng, S. Xu and X. Wu, "Object detection with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, 2019.
[23] Y. Liu, L. Jin, Z. Xie, C. Luo, S. Zhang *et al.,* "Tightness-aware evaluation protocol for scene text detection," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, pp. 9612–9620, 2019.
[24] A. Simon, J. -C. Pret and A. P. Johnson, "A fast algorithm for bottom-up document layout analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, pp. 273–277, 1997.
[25] K. Papineni, S. Roukos, T. Ward and W. J. Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, pp. 311–318, 2002.
[26] L. Jinhyuk, Y. Wonjin, K. Sungdong, K. Donghyeon, K. Sunkyu *et al.,* "BioBERT: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.
[27] Y. Wu, M. Schuster, Z. Chen, Q. Le, M. Norouzi *et al.,* "Google's neural machine translation system: Bridging the gap between human and machine translation," arXiv preprint arXiv:1609.08144, 2016. [Online]. Available: https://arxiv.org/abs/1609.08144v2.

Appendix A

Journals	Numbers of Selected Papers
Train Set	
Journal of the Korean Cleft Palate-Craniofacial Association	248
Korean Journal of Construction Engineering and Management	575
Journal of Internet Computing and Services	513
Journal of the Korean Society of Radiology	726

(Continued)

Continued

Journal of the Korea Institute of Information and Communication Engineering	2477
Korean Journal of Materials Research	892
Food Science of Animal Resources	636
Korean Journal of Pediatrics	812
Korean Chemical Engineering Research	761
Journal of Information Processing Systems	97
Journal of Digital Convergence	3351
Journal of the Korea Convergence Society	1411
Journal of the Korean Society of Clothing and Textiles	621
Molecules and Cells	360
Journal of Korean Academy of Nursing	772
Journal of the Korean Society of Integrative Medicine	310
The Journal of The Institute of Internet, Broadcasting and Communication	1558
JOURNAL OF KOREA WATER RESOURCES ASSOCIATION	727
Bulletin of the KMS	1133
International journal of advanced smart convergence	341
Archives of Plastic Surgery	1135
The Journal of The Korea Institute of Electronic Communication Sciences	852
Applied Chemistry for Engineering	537
Journal of Korea Institute of Information, Electronics, and Communication Technology	448
Pediatric Gastroenterology, Hepatology & Nutrition	373
The Journal of the Korea Contents Association	5524
The Journal of Korean Oriental Internal Medicine	603
KSII Transactions on Internet and Information Systems (TIIS)	1256
Journal of Korean Medicine Rehabilitation	250
The Korean Journal of Physiology and Pharmacology	518
The Journal of Korean Physical Therapy	561
Journal of Preventive Medicine and Public Health	420

(Continued)

Continued

The Korean Journal of Thoracic and Cardiovascular Surgery	779
Child Health Nursing Research	393
Biomolecules & Therapeutics	580
Asian-Australasian Journal of Animal Sciences	417
JOURNAL OF ENVIRONMENTAL HEALTH SCIENCES	533
The Korean Journal of Parasitology	705
Journal of the Korean Society of Physical Medicine	547
Journal of The Korea Institute of Information Security and Cryptology	919
Journal of the Korean Association for Science Education	638
Journal of the Korean Applied Science and Technology	747
Journal of the Korean Society of Civil Engineers	1226
Journal of Digital Contents Society	580
The Korean Journal of Food and Nutrition	967
Journal of Korean Neurosurgical Society	1132
Journal of Microbiology and Biotechnology	1281
The Korean Journal of Applied Statistics	591
Journal of Power Electronics	154
Microbiology and Biotechnology Letters	463
The Journal of Korean Medicine Ophthalmology & Otorhinolaryngology & Dermatology	418
Journal of the Korean Library and Information Science Society	477
Journal of The Korea Society of Computer and Information	1912
The Journal of Oriental Obstetrics & Gynecology	404
The Transactions of the Korean Institute of Electrical Engineers	787
Journal of Life Science	1203
ETRI Journal	1100
KIPS Transactions on Software and Data Engineering	523
Journal of Korean Navigation and Port Research	463

(Continued)

Continued	
JOURNAL OF KOREA MULTIMEDIA SOCIETY	939
Sub Total	51676
Test Set	
The Journal of Korean Academy of Prosthodontics	444
Korean Journal of Food Science and Technology	869
Nuclear Engineering and Technology	755
BMB Reports	647
Journal of Korean Society of Dental Hygiene	873
Journal of Korea Academia-Industrial cooperation Society	7109
JOURNAL OF BROADCAST ENGINEERING	667
Journal of the Korea Society Industrial Information System	568
Journal of IKEEE	779
Journal of Convergence for Information Technology	620
Sub Total	13331
Total	65007