

# Towards Comprehensive Lecture Slides Understanding: Large-scale Dataset and Effective Method

Enming Zhang<sup>1</sup>, Yuzhe Li<sup>1</sup>, Yuliang Liu<sup>1</sup>, Yingying Zhu<sup>1\*</sup>, Xiang Bai<sup>1</sup>

<sup>1</sup>Huazhong University of Science and Technology  
{emzhang,yzli,ylliu,yyzhu,xbai}@hust.edu.cn

## Abstract

*Online education has been widespread in worldwide universities and educational institutions. Lecture slides, a fundamental component of online education, contain a wealth of information, playing a crucial role in learning. However, previous works have not yet paid sufficient attention to understanding lecture slides, including the absence of the large-scale dataset and comprehensive understanding tasks. To facilitate the research about lecture slides understanding, we establish the LecSlides-370K, which consists of 25,542 lectures with 370,078 slides across 15 areas. We also introduce two comprehensive tasks, Lecture Summary and Lecture Question Answering (QA), for providing different perspectives of slides understanding. Furthermore, complex and flexible text relations can hinder the understanding of the internal logic of slides. To address this challenge, we propose a novel method, named SlideParser, which includes an auxiliary branch to predict text relations within slides and enhance attention between related texts, thereby improving slides understanding. With extensive experiments, we show the superiority of our proposed method on both LecSlides-370k and SlideVQA. Dataset and Codes are available at [https://github.com/zamling/LecSlides\\_370K](https://github.com/zamling/LecSlides_370K)*

## 1. Introduction

With the rapid progress in online education, an increasing number of universities and educational institutions have adopted online learning in many platforms [20, 21, 25]. Among diverse online education formats, such as lecture videos and live streams, slides often serve as a key medium for conveying significant knowledge. Consequently, more researchers have turned their attention to this field. However, there are two obstacles to understand lecture slides: 1) Texts in slides are often conclusive and headline-style statements, which complicate the task of providing detailed

content. 2) The complex text relations pose additional challenges for models attempting to grasp the internal logic and generate reasonable responses.

Recently, several works [4, 5, 7, 14, 17, 27] have noticed the importance of lecture slides and introduced corresponding datasets, such as text detection in slides [5], slide segmentation [7] and image-text retrieval for slides and corresponding speech [14]. However, none of them has yet paid attention on comprehensive understanding slides. SlideVQA [27], a recent slides dataset, provides a Question-Answering (QA) benchmark for slides, yet it still overlooks the global understanding task, such as generating a detailed summary for a complete set of slides. Additionally, the restricted scales and limited number of lecture areas also constrain the development of lecture slides understanding.

Aiming to address the limitations in the lecture slides dataset, we establish a large-scale and multi-task dataset for comprehensive slides understanding, named LecSlides-370K. Specifically, we collect 25,542 lectures composed of 370,078 slides. These lectures span 15 lecture areas, including business, natural sciences, computer sciences, and so on, significantly surpassing previous datasets in both scale and diversity, as shown in Fig. 1 (a). Furthermore, we introduce two tasks based on our dataset, termed Lecture Summary and Lecture Question Answering (QA). These two tasks represent the most frequently utilized functions for students, offering varied perspectives for slides understanding. Specifically, Lecture Summary focuses on extracting key information and fostering a global understanding, while Lecture QA focuses on locating problems and understanding fine-grained content. Finally, our dataset includes 114,472 summary annotations and 259,461 QA pairs.

Based on LecSlides-370K dataset, we evaluate the existing NLP methods [15, 22], document understanding methods [10, 28, 32, 33] and vision-language models [16, 34]. The results reveal the inferior performance of these methods. We argue that this inferiority can be attributed to the fact that these methods have not yet adequately considered the unique characteristic of lecture slides, that text relations in slides are complex and contain a wealth of logic and rel-

\*Corresponding author.

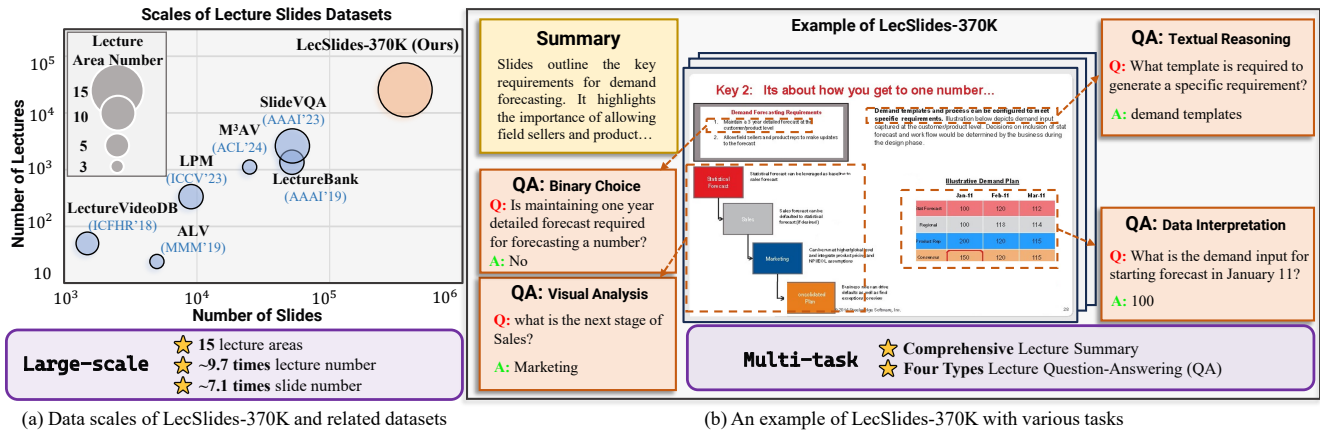


Figure 1. LecSlides-370K is a large-scale and multi-task lecture slides dataset. It contains 25,542 lectures and 370,078 slides across 15 lecture areas, which has around 9.7 times lecture numbers and 7.1 times slide numbers compared with SlideVQA [27]. (a) A comparison between LecSlides-370K and other slides dataset in terms of the number of lectures, slides and lecture areas. The detailed information is shown in Table 1. (b) An example of LecSlides-370K, including Lecture Summary and Lecture QA tasks.

evance. For instance, as shown in Fig. 1 (b), texts displayed in the upper left corner indicate detailed requirements for forecasting, while texts on the right side of each block explain each stage in the flow chart. Therefore, understanding complex text relations in slides can grasp the internal logic, and is the key part of understanding slides.

To extract the complex text relations and enhance slides understanding, we propose a novel method called SlideParser. The key of SlideParser is how to effectively predict and utilize complex text relations in slides. Specifically, we propose an auxiliary branch, termed Text Relation Prediction Module, comprising the Relation Prediction Head (RPH) to predict relations among texts and the Relation Attention Layer (RAL) to leverage predicted text relations, enhancing attention between related texts while reducing it between unrelated texts. Additionally, to enhance the quality of predictions, we build a weak supervision framework for RPH with a pseudo relation label generation pipeline, named DL-Cluster, which considers both text spatial and semantic relations, yielding superior text relation labels compared to traditional clustering methods [2, 6, 9].

In our experiments on LecSlides-370K, related baselines struggle to extract the main point of lecture slides and pinpoint the correct content corresponding to the question, revealing the challenges of our dataset. By effectively predicting and utilizing text relations, SlideParser outperforms all the related methods in the LecSlides-370K Lecture Summary and Lecture QA benchmarks. Additionally, we evaluate SlideParser on the SlideVQA [27] dataset, further validating the effectiveness of our text relation design.

In conclusion, our contributions are:

- We establish a large-scale and multi-task slides understanding dataset, termed LecSlides-370K, comprising 25,542 lectures across 15 lecture areas and 370,078

slides. We also introduce two comprehensive tasks, Lecture Summary and Lecture QA.

- We propose a novel method, SlideParser, including an auxiliary branch to effectively predict and utilize complex text relations in slides, enhancing slides understanding.

## 2. Related Work

### 2.1. Lecture Slides Datasets

Recently, several datasets based on lecture slides have been proposed, which are most relevant to our current work, including the LectureBank Dataset [17], ALV [7], LectureVideoDB [5], LPM Dataset [14], M<sup>3</sup>AV [4] and SlideVQA [27], as shown in Table 1.

LectureBank [17] is annotated for prerequisite relationships among topics, providing a structure of connected knowledge. ALV dataset [7] primarily focuses on segmenting the entire video lecture. LectureVideoDB is formulated to investigate text detection and recognition in video lectures. Furthermore, LPM [14] attempts to explore the retrieval of spoken words with corresponding visual cues. SlideVQA, a recent work for slides, proposes a Visual Question Answering (VQA) task on it.

In summary, previous datasets contain two primary limitations: 1) Most of datasets do not pay sufficient attention to slides understanding, and the tasks they define also do not provide multiple perspectives for understanding. 2) The limited data scale hinders the advancement of understanding lecture slides. Therefore, a substantial requirement in the research community emerges for a large-scale slides dataset with comprehensive understanding tasks.

Dataset	Size		Annotation			Task
	Lectures	Slides	OCR	QA	Summary	
LectureVideoDB [5]	24	5,000	✓	-	-	Text Detection
LectureBank [17]	1,352	51,939	✓	-	-	Prerequisite Chain
ALV Dataset [7]	-	1,498	✗	-	-	Slide Segmentation
LPM [14]	334	9,031	✓	-	-	Image Retrieval
M <sup>3</sup> AV [4]	1113	24,956	✓	-	-	ASR & TTS
SlideVQA [27]	2,619	52,380	✓	14,500	-	QA
<b>LecSlides-370K (Ours)</b>	<b>25,542</b>	<b>370,078</b>	<b>✓</b>	<b>259,461</b>	<b>114,472</b>	<b>QA &amp; Summary</b>

Table 1. Data scales, annotations and tasks for different lecture slides datasets. LecSlides-370K contain the largest data scale, the most comprehensive annotations and two understanding tasks to evaluate slides understanding from multiple perspectives.

## 2.2. Slides Understanding Methods

Although there is a lack of specific methods for slide understanding, other area methods can be used in slide understanding, mainly including three categories: vision-language methods [16, 31, 34, 36], NLP methods [15, 22], and document understanding methods [10, 28].

Vision-language methods typically begin with large-scale pre-training tasks and then fine-tune on downstream multi-modal understanding tasks. These methods often integrate a vision-language fused transformer encoder [16, 31, 36]. NLP methods provide a plain text solution, such as T5 [22] and BART [15], following an encoder-decoder architecture to process textual contents and generate textual outputs. Furthermore, inspired by NLP approaches, document understanding methods replace traditional transformer encoder to multi-modal encoder, processing and fusing textual and visual content. For example, LayoutLM series [10, 32, 33] integrate text, image features, and 2D positional embeddings in the multi-modal encoder.

However, these related methods do not consider the complex text relations in slides, a crucial part of slides understanding, thereby yielding inferior performance.

## 3. LecSlides-370K Dataset

### 3.1. Data Collection

To build a large-scale and multi-modal lecture slides dataset, we collect slides from Slideshare\*. Our raw data consists of 720,673 slides from 39,784 lectures at the beginning. However, some of these slides are irrelevant to the lecture content, such as advertisements, or contain extraneous information, which is not essential for comprehension. Thus, we filter these slides by their contents. Finally, LecSlides-370K contains 370,078 slides from 25,542 lectures across 15 areas, such as computer science, marketing, medicine, and so on. As shown in Table 1, compared with

the previous largest datasets, SlideVQA [27], LecSlides-370K has around 9.7 times lecture numbers and 7.1 times slide numbers. The superiority of data scale and diversity renders our dataset more comprehensive and challenging.

Moreover, since the length of these raw lecture slides varies widely, from a few to hundreds, which makes it difficult for researchers to use them, we segment these raw slides into 114,472 lecture chunks, following the principle that each chunk belongs to the same sub-topic. For instance, if one slide initially displays the first point of a list, and the subsequent slide reveals the following points, these slides should belong to the same chunk. The distribution of slides number in each lecture chunk is shown in Fig. 2 (b). Additionally, we annotate the positions of texts, illustrations, and tables in slides to facilitate the study and encourage researchers to focus more on developing new methods rather than information extraction and data processing. Furthermore, we also annotate the textual contents in slides, which can be utilized by the models to understand slides better.

### 3.2. Task Definition

We define two tasks: Lecture Summary and Lecture QA. These two tasks not only involve most application scenarios, but also take into account the global and detailed understanding of slides, fully reflecting the understanding ability of models. In the context of Lecture Summary, given a lecture chunk  $\mathbf{I} = \{I_1, I_2, \dots, I_j\}^M$ , comprising  $M$  slide images, the model’s objective is to analyze the content of each slide, integrate contextual information, retain key details, and eliminate redundancies. Finally, the model outputs a summary of all slides. Contrary to the global scope of Lecture Summary, Lecture QA primarily concentrates on locating and comprehending specific content within a lecture chunk. Given a lecture chunk and a query  $q$ , the model need to identify the slide corresponding to the question and then provide an answer based on the content of that slide. QA serves to assess the model’s fine-grained understanding of slides. In total, We annotate 114,472 summary labels and

\*<https://www.slideshare.net>

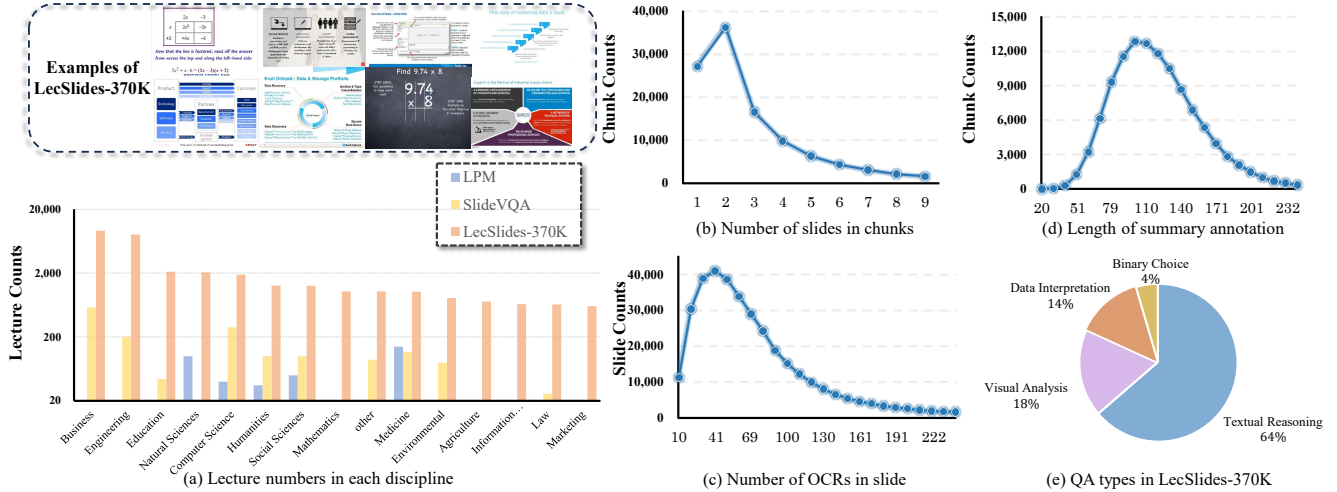


Figure 2. Examples and statistics of LecSlides-370K. (a) Lecture numbers in each discipline for LecSlides-370K, SlideVQA [27] and LPM [14]. (b) The number of slides for each chunk. (c) The distribution of OCRs in slides. (d) The distribution of summary length. (e) The percentage of each QA type.

259,461 QA pairs.

### 3.3. Dataset Statistics and Analysis

In this section, we aim to provide a comprehensive analysis and understanding of the LecSlides-370K dataset.

#### 3.3.1. Lecture Diversity

We collect slides from various disciplines, as shown in Fig. 2 (a). There are 15 disciplines, and the highest proportion of data pertains to business, while law and marketing lectures constitute a relatively small percentage. Compared with previous slides datasets, such as SlideVQA [27] and LPM [14], LecSlides-370K contains more disciplines, such as marketing, agriculture, and so on. The diversity of disciplines contains rich teaching content, bringing challenges that models need to handle more diverse scenes.

#### 3.3.2. Professionalism

Lecture slides usually contain specialized knowledge and terminologies, bringing more challenges. Therefore, we use the Flesch Reading Ease (FRES) score [13] to evaluate the difficulty of contents and get an average score of 34.3 under this metric. This indicates that the content of our dataset is on **‘Hard to Read’** level. Moreover, we also use the Flesch–Kincaid Grade Level [13] to reflect which education level our dataset is appropriate for and obtain a 14.1 grade on average. These show there is a professional difficulty in our dataset.

#### 3.3.3. Slide Image

For LecSlides-370K, there are 114,472 lecture chunks corresponding to 370,078 slide images. Each chunk contains a varying number of slide images, with an average of 3.2 images per chunk. The distribution of chunks containing

different numbers of slide images is shown in Fig. 2 (b), and most chunks contain two slide images. Moreover, slide images are rich in visual information. By our count, there are 94,906 diagrams, 34,562 illustrations, and 18,938 tables in LecSlides-370K dataset. We further show the statistics for textual content in slides. We totally annotate 28,220,274 words, with an average of 76.2 words. The size of the vocabulary for our dataset is 84,630. In Fig. 2 (c), we show the distribution of texts across slides, revealing that the majority of slides contain fewer than 100 words.

#### 3.3.4. Characteristics of Annotation

For summary annotations, Fig. 2 (d) illustrates the length distribution of the annotations. The average length of the summary is 124.6 words. We further comprehensively analyze QA annotations, including four distinct question types: binary choice, visual analysis, data interpretation and textual reasoning. As shown in Fig. 2 (e), the highest proportion of question types is textual reasoning, while binary choice constitutes the smallest percentage. This diverse set of question types covers the whole slide content, enabling a comprehensive evaluation of the model’s QA capabilities. In addition, the average length of answer is 4.1 words.

### 3.4. Evaluation Protocol

LecSlides-370K dataset contains 109,472 slide chunks for the training set, 2,000 chunks for the validation set and 3,000 chunks for the test set. We adapt the ROUGE-1, ROUGE-L [18] and METEOR [3] as evaluation metrics for Lecture Summary task. These metrics are widely used in image caption and text generation, as they consider the quality of generated texts in different aspects. Following SlideVQA [27], we choose EM (Exact Matching) and F1

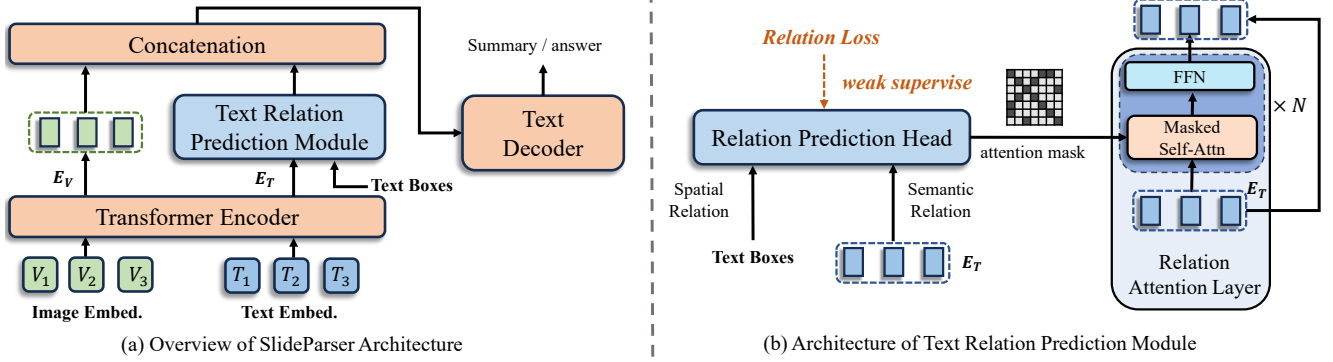


Figure 3. (a) The architecture of our proposed method, SlideParser. It consists of a Transformer Encoder, a Text Relation Prediction Module, and a Text Decoder. (b) The details of Text Relation Prediction Module, which can predict and utilize text relations effectively, including the Relation Prediction Head (RPH) with weak supervision and the Relation Attention Layer (RAL).

score as evaluation metrics for the Lecture QA task.

## 4. Method

As shown in Fig. 3 (a), SlideParser consists of a Transformer Encoder, a Text Relation Prediction Module, and a Text Decoder. Texts and image patches from slides are fed into the Transformer Encoder, and the output text embeddings are forwarded to the Text Relation Prediction Module to predict text relations and utilize them to update text embeddings. Finally, updated text embeddings combine with output image embeddings and are fed into the Text Decoder to generate the summary or answer.

### 4.1. Input Schema

SlideParser is built on a unified text-image multi-modal Transformer Encoder to learn cross-modal representations. Similar to previous document understanding works [10, 28], the Transformer Encoder adopts a multi-layer structure, with each layer primarily comprising multi-head self-attention and feed-forward networks. The input of Transformer Encoder consists of text embeddings  $\mathbf{X} = x_{1:L}$  and image embedding  $\mathbf{V} = v_{1:M}$ , where  $L$  and  $M$  are sequence length of textual and visual embeddings respectively. Specifically, for text embedding  $\mathbf{X}$ ,  $L$  text tokens  $\{s_i\}_{i=1}^L$  are encoded by word embedding with 1D position and 2D layout position embedding [28]. Meanwhile, the slide image  $\mathbf{I} \in \mathbb{R}^{H \times W \times C}$  is divided into  $M = \frac{H}{P} \times \frac{W}{P}$  image patches. These patches are then processed through Patch Embedding to generate the image embeddings,  $\mathbf{V}$ .

### 4.2. Text Relation Prediction Module

In Transformer Encoder, 2D position embedding, which can be used to comprehend layout and text relations, has been demonstrated in previous document understanding works [10, 28]. However, in the context of slides, text relations tend to be more complex compared to traditional doc-

uments, such as examples in Fig. 2. Merely incorporating 2D position embeddings into the transformer encoder may not suffice for comprehensive relation comprehension and could potentially introduce inaccuracies.

To address this issue, we introduce the Text Relation Prediction Module, including the Relation Prediction Head (RPH) with weak supervision and the Relation Attention Layer (RAL), as shown in Fig. 3 (b). This module predicts text relations by RPH and utilizes relations to enhance attention between related texts while reducing attention between unrelated texts by RAL. Additionally, to improve the quality of predicted relations, we establish a weak supervision pipeline. We will introduce each block in detail.

#### 4.2.1. Relation Prediction Head

To predict the accurate text relations, inspired by [30], RPH leverages text position and semantic information to predict text relations. Given text bounding boxes  $\{b_i\}_{i=1}^L$ , we define the box delta as:

$$\Delta(b_i, b_j) = \{(x_i^{ctr} - x_j^{ctr})/w_i, (y_i^{ctr} - y_j^{ctr})/h_i, \log(w_i/w_j), \log(w_i/w_j)\} \quad (1)$$

where  $x_i^{ctr}$  and  $y_i^{ctr}$  represent the position of the center point for bounding box, and  $w$  and  $h$  are the width and height of box respectively. Moreover, we present the spatial relation of two boxes as:  $\mathbf{r}_{b_i, b_j} = \{\Delta(b_i, b_j), \Delta(b_i, b_{i,j}), \Delta(b_{i,j}, b_j)\}$ , which is a 12-D vector. Specifically,  $b_{i,j}$  is a union bounding box for  $b_i$  and  $b_j$ . Then, we employ an *MLP layer* and integrate all box relations to get the spatial relation matrix  $\mathbf{R}_{\text{spa}} \in \mathbb{R}^{L \times L}$ . In addition, RPH also considers semantic relations. Given output text embeddings from Transformer Encoder,  $\mathbf{E}_T \in \mathbb{R}^{L \times D}$ , we compute the semantic relation between two texts as

$$\mathbf{R}_{\text{sem}} = (\mathbf{W}_q \mathbf{E}_T \times \mathbf{W}_k \mathbf{E}_T^T) / \sqrt{D} \quad (2)$$

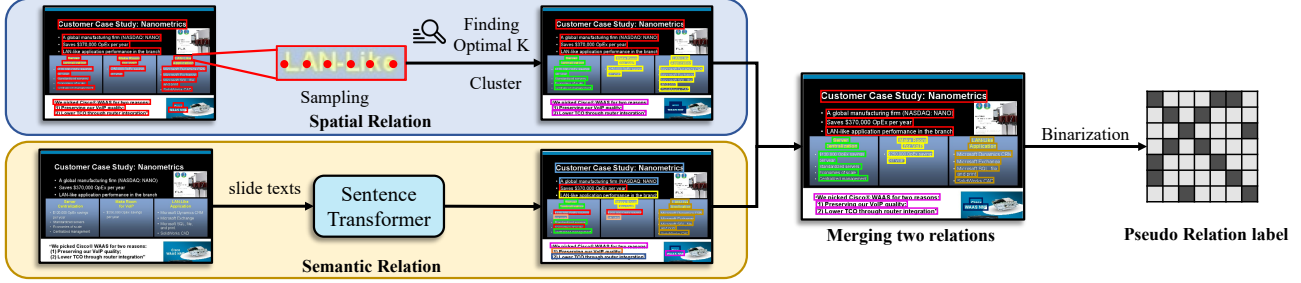


Figure 4. The Pipeline of DL-Cluster to generate pseudo relation label. We combine spatial and semantic text relations to get pseudo labels.

The predicted relation matrix  $\hat{M}_R \in \mathbb{R}^{L \times L}$  is computed as

$$\hat{M}_R = \text{Sigmoid}(\mathbf{R}_{\text{spa}} + \mathbf{R}_{\text{sem}}) \quad (3)$$

#### 4.2.2. Weak Supervised Label Generation

In our experiment, we find that the RPH struggles to generate reasonable text relations without the presence of relation supervision. As a solution, we introduce a weak supervision pipeline to enhance the quality of predictions. Previous works [1, 8] rely on traditional cluster methods [6, 9] to predict text relations. However, these traditional methods can not handle diverse layout format in slides effectively. Moreover, our observations suggest that both spatial and semantic relations play crucial roles in slides.

Thus, we introduce a dual-stream pseudo relation label generation pipeline, termed DL-Cluster (Dynamic Layout Cluster), as shown in Fig. 4. To obtain spatial relations, we first sample  $N$  reference points as cluster points for each text box. Because of the diverse nature of text relations in slides, a predefined number of clusters may not be ideal for accurate relation predictions. Hence, we opt for a dynamic approach to determine the cluster number for each slide.

Specifically, based on Kmeans++ [2], we iterate over the number of cluster  $k$  from 2 to  $\mathcal{K}$  and determine the optimal cluster number by Silhouette method [24]. Silhouette analyzes the distances of each cluster point to its own cluster and its closest neighboring cluster. The optimal  $k$  is obtained when the average distance of a cluster point to all the other points in its own cluster is minimum, while the average distance of a point to the nearest neighboring cluster’s points is maximum. In our experiment,  $\mathcal{K} = 8$ .

Subsequently, we assign a cluster ID to each box through a voting mechanism among reference points. Then, the spatial relation score  $s_{\text{spa}}$  is computed as.

$$s_{\text{spa}} = \begin{cases} 1 - \frac{1}{N} \sum_{\varepsilon=1}^N \|x_{i,\varepsilon} - \mu_j\|, & \text{when } C_i = C_j \\ 0, & \text{when } C_i \neq C_j \end{cases} \quad (4)$$

where  $C_i$  and  $C_j$  are clusters and  $\mu_j$  is cluster centroid of  $C_j$ . Specifically, scores are assigned based on the average distance between reference points and the cluster centroid

when these texts are deemed related and are placed within the same cluster. Conversely, when texts are considered unrelated, the score is set to zero.

Simultaneously, to obtain semantic relation, we extract textual contents in each text box, and compute the semantic similarity by the cosine distance of Sentence Transformer [23] and get the semantic relation score  $s_{\text{sem}}$ . These two relation scores are combined as

$$\mathbf{s} = \beta_1 \text{Norm}(\mathbf{s}_{\text{spa}}) + \beta_2 \text{Norm}(\mathbf{s}_{\text{sem}}) \quad (5)$$

we set the default values of  $\beta_1 = 0.7$ ,  $\beta_2 = 0.3$ . Finally, the relation matrix  $M_R = \{s_{ij}\}^{L \times L}$  is binarized with 0.5 threshold. Then, the text relation loss can be calculated as

$$\mathcal{L}_{\text{rel}} = \text{BCELoss}(\hat{M}_R, M_R) \quad (6)$$

#### 4.2.3. Relation Attention Layer

With the predicted text relations, we utilize relations by Self Attention [29], as shown in Fig. 3 (b). Specifically, for predicted relation matrix  $\hat{M}_R$ , we generate attention mask by assigning  $-\infty$  when  $m_{ij} < 0.5$ ,  $m_{ij} \in \hat{M}_R$ . Subsequently, the output text embedding  $\mathbf{E}_T$  are fed into N-layer self-attention blocks with this attention mask, resulting in the updated embeddings,  $\mathbf{E}'_T$ . It enhances attention between related texts and isolates attention between unrelated texts. Finally, we combine  $\mathbf{E}_T$  and  $\mathbf{E}'_T$  by residual connection.

### 4.3. Text Decoder

Following previous work [28], we adapt a T5 decoder as our text decoder, a uni-directional Transformer decoder that performs the next prediction task to generate texts, such as summary or answer. We choose Cross Entropy loss to supervise text generation. The total loss of SlideParser is

$$\mathcal{L} = \mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{rel}} \quad (7)$$

## 5. Experiments

### 5.1. Implement details

For SlideParser, both the Transformer Encoder and the Text Decoder are initialized from the pre-trained UDOP [28]

Method	#Params	val			test		
		R-1	R-L	METEOR	R-1	R-L	METEOR
BART-base [15]	217M	33.1	21.1	21.9	32.8	20.1	21.5
BART-large [15]	509M	35.9	21.9	24.2	35.0	21.0	22.9
T5-base [22]	272M	34.6	21.6	22.8	34.3	20.7	22.4
T5-large [22]	803M	36.2	22.1	24.9	35.1	21.3	23.2
Vid2Seq [34]	314M	16.0	11.8	12.9	17.8	13.1	14.1
mPLUG-base [16]	350M	27.1	19.0	21.6	27.6	19.3	21.7
mPLUG-large [16]	600M	30.0	20.5	24.0	29.8	20.3	23.6
Donut [12]	543M	27.4	18.6	21.3	27.6	18.5	21.1
LayoutLMv3-base [10]	533M	33.6	21.0	23.8	31.7	19.8	22.6
LayoutLMv3-large [10]	768M	34.6	21.6	24.8	33.9	21.0	23.2
UDOP [28]	794M	36.1	22.1	26.4	35.3	21.5	25.6
SlideParser (Ours)	810M	<b>42.2</b>	<b>25.3</b>	<b>31.3</b>	<b>41.7</b>	<b>24.7</b>	<b>30.4</b>

Table 2. Performance on Lecture Summary in LecSlides-370K

Method	#Params	val		test	
		EM	F1	EM	F1
BART-base [15]	217M	18.9	25.1	18.5	24.7
BART-large [15]	509M	20.6	27.1	19.8	26.3
T5-base [22]	272M	19.1	26.0	18.7	25.3
T5-large [22]	803M	20.9	27.4	20.0	27.1
Vid2Seq [34]	314M	7.0	14.3	6.7	13.6
mPLUG-base [16]	350M	8.4	10.9	8.5	10.1
mPLUG-large [16]	600M	9.5	14.3	9.5	14.2
LayoutLMv3-base [10]	533M	12.5	15.8	12.4	15.1
LayoutLMv3-large [10]	768M	13.6	19.3	13.5	19.1
UDOP [28]	794M	19.2	26.0	18.7	25.6
SlideParser (Ours)	810M	<b>23.7</b>	<b>33.0</b>	<b>23.6</b>	<b>32.1</b>

Table 3. Performance on Lecture QA in LecSlides-370K

model weights. We adapt AdamW [19] optimizer with  $\beta_1$  of 0.9, the  $\beta_2$  of 0.999, and the weight decay of 0.02. We use the cosine learning rate scheduler and the initial learning rate is  $4 \times 10^{-5}$ . The total number of iterations is 41K. Moreover, we choose a simple concat operator for encoder outputs to handle multiple slide situations for all models.

To provide a full view of LecSlides-370K, we evaluate other related methods from various domains: NLP models such as T5 [22] and BART [15], vision-language models such as mPLUG [16] and Vid2Seq [34], and document understanding models such as LayoutLMv3 [10], Donut [12] and UDOP [28]. Specifically, LayoutLMv3 lacks a text decoder, which is unable to handle text generation tasks. To address this limitation, we incorporate the same text decoder as our proposed method.

## 5.2. Results

### 5.2.1. Performance on LecSlides-370K

At the beginning, we analyze Lecture Summary task in LecSlides-370K. As shown in Table 2, SlideParser achieves notable scores of 42.2 in ROUGE-1, 25.3 in ROUGE-L, and 31.3 in METEOR. This performance surpasses other baseline methods, attributed to its effective understanding text relations. Specifically, the methods without processing tex-

Model Name	dev		test	
	EM	F1	EM	F1
VinVL [36]	9.4	11.4	10.7	13.5
PreasM [35]	36.3	41.9	30.7	38.2
T5 [22]	35.2	41.3	31.0	39.7
LayoutT5 [26]	38.9	44.8	31.7	39.9
LayoutLMv2 [33]	26.5	33.4	21.4	29.3
FiD [11]	37.6	42.9	30.4	38.9
M3D [27]	41.3	47.1	33.5	41.7
SlideParser (Ours)	<b>44.1</b>	<b>49.8</b>	<b>37.5</b>	<b>43.9</b>
SlideParser (Ours) <sup>†</sup>	49.8	53.9	42.4	47.7

Table 4. Performance on SlideVQA benchmark. † represents pre-training on LecSlides-370K and fine-tuning on SlideVQA.

tual modality, including Vid2Seq [34], mPLUG series [16] and Donut [12], exhibit the worst results. We argue that only processing visual modality may lead to a lack of detailed content extraction in slides. In addition, NLP methods, such as T5 [22] and BART [15], perform relatively well but still lag behind our proposed method, containing excellent multi-modal fusion and text relation utilization.

In contrast to Lecture Summary, which primarily evaluates global understanding of slides, Lecture QA task measures methods for in-depth understanding of specific content. As shown in Table 3, SlideParser achieves superior performance compared to other baselines. Furthermore, visual-language models such as mPLUG and Vid2Seq still perform the worst among all baselines. It shows that detailed text information is still crucial for QA task.

### 5.2.2. Performance on SlideVQA

To further evaluate our proposed method in QA task, we also fine-tune and evaluate SlideParser on SlideVQA, a VQA benchmark in multi-image slides. As shown in Table 4, SlideParser achieves the best performance compared with other methods, which further demonstrates the effective designs of SlideParser. In an extensive experiment, we pre-train SlideParser in our dataset and fine-tune on SlideVQA, denoted as SlideParser<sup>†</sup>. Results show that large-scale and comprehensive LecSlides-370K yield a great performance enhancement, demonstrating our dataset can also serve as a valuable large-scale pre-training dataset.

## 5.3. Ablation Study

To gain a deeper analysis of SlideParser, we conduct ablation experiments in LecSlides-370K Lecture Summary task.

### 5.3.1. Effect of Text Relation Prediction Module.

We first evaluate the impact of incorporating the Text Relation Prediction Module. As shown in Table 5, results demonstrate the removal of the entire Text Relation Prediction Module (in r1) yields the poorest performance. Furthermore, as we add RPH and RAL to predict and utilize rela-

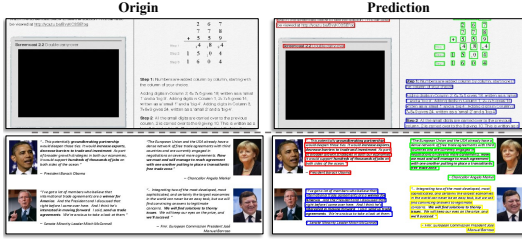


Figure 5. Visualization of predicted text relations in RPH

	RPH	RAL	Pseudo Label	R-1	R-L	METEOR
r1	✗	✗	✗	36.3	22.6	26.9
r2	✓	✓	✗	37.5	23.2	27.7
r3	✗	✓	✓	41.8	25.0	30.8
r4	✓	✓	✓	<b>42.2</b>	<b>25.3</b>	<b>31.3</b>

Table 5. Ablation on Text Relation Prediction Module

Cluster Type	R-1	R-L	METEOR
Kmeans (k=3)	40.9	24.8	29.8
Kmeans (k=4)	40.7	24.7	29.6
HAC(n=3)	41.0	24.8	30.1
DBSCAN	41.3	24.8	30.6
<b>DL-Cluster</b>	<b>42.2</b>	<b>25.3</b>	<b>31.3</b>

Table 6. Ablation on cluster types in pseudo label generation

tions without any supervision (in r2), the performance only has a slight improvement. It demonstrates the RPH struggles to converge effectively in generating reasonable text relations. In addition, we also explore the approach of directly feeding pseudo relation labels generated by DL-Cluster into the subsequent RAL, instead of utilizing learnable relations (in r3). Although it exhibits substantial improvement, yet it continues to fall short in comparison to leveraging learnable text predictions and establishing weak supervision (r3 vs r4). It shows that learnable relations could be refined during the training process.

Moreover, we provide visualization results in Fig. 5. We assign the same color for two texts if their relation score is larger than 0.5. The results illustrate that the RPH can predict reasonable text relations under weak supervision.

### 5.3.2. Different Methods for Pseudo Label Generation.

Traditional cluster methods are widely used in previous works for relation prediction [1, 8]. Therefore, we compare our proposed DL-Cluster with other cluster methods. As shown in Table 6, DL-Cluster surpasses clustering techniques with predefined cluster numbers like K-means and Hierarchical Agglomerating Clustering (HAC). Additionally, DL-Cluster outperforms methods without specified cluster numbers, such as DBSCAN [6], due to its capacity to consider spatial and semantic aspects concurrently.

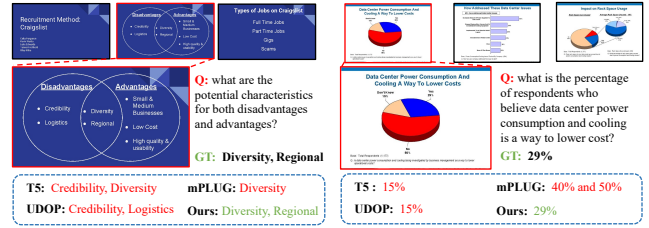


Figure 6. Qualitative results of Lecture QA in LecSlides-370K.

Spatial	Semantic	R-1	R-L	METEOR
✗	✓	38.3	23.7	28.5
✓	✗	41.4	24.8	30.7
✓	✓	<b>42.2</b>	<b>25.3</b>	<b>31.3</b>

Table 7. Ablation on designs of DL-Cluster

### 5.3.3. Design of DL-Cluster.

We further investigate the impact of text relations represented in spatial and semantic dimensions. As shown in Tab. 7, removing either of these dimensions leads to a decline in performance, and spatial relations have a larger impact. The best results are achieved when both aspects were used simultaneously, indicating that both dimensions complement each other to enhance overall performance.

## 5.4. Visualization

In a side-by-side qualitative analysis for Lecture QA in LecSlides-370K, we compare our proposed method with other baselines in related fields. As shown in Fig. 6, results show that other methods face two challenges: 1) They can not grasp correct internal logic due to the complex text relations. 2) As the number of diagrams on slides increases, other methods struggle to pinpoint the correct diagram corresponding to the question. In contrast, SlideParser excels in accurate responses by comprehending text relations effectively and easily determining the answer’s location within the slides.

## 6. Conclusion

In this work, we introduce a large-scale and multi-task lecture slides dataset, LecSlides-370K, which has a larger data scale and more diverse disciplines. Furthermore, considering the pivotal role of text relation prediction and utilization for slides understanding, we propose SlideParser. The results demonstrate the effect of enhanced text relation design, and our method outperforms other models. We believe LecSlides-370K and SlideParser can prompt lecture slides research.

## Acknowledgments

This work was supported by the NSFC (62206103 and 62225603)

## References

- [1] Rhys Agombar, Max Luebbing, and Rafet Sifa. A clustering backed deep learning approach for document layout analysis. In *Machine Learning and Knowledge Extraction: 4th IFIP TC 5, TC 12, WG 8.4, WG 8.9, WG 12.9 International Cross-Domain Conference, CD-MAKE 2020, Dublin, Ireland, August 25–28, 2020, Proceedings 4*, pages 423–430. Springer, 2020. 6, 8
- [2] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *Society for Industrial and Applied Mathematics*, page 1027–1035, 2007. 2, 6
- [3] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 4
- [4] Zhe Chen, Heyang Liu, Wenyi Yu, Guangzhi Sun, Hongcheng Liu, Ji Wu, Chao Zhang, Yu Wang, and Yanfeng Wang. M3av: A multimodal, multigenre, and multipurpose audio-visual academic lecture dataset. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 9041–9060, Bangkok, Thailand, 2024. Association for Computational Linguistics. 1, 2, 3
- [5] Kartik Dutta, Minesh Mathew, Praveen Krishnan, and CV Jawahar. Localizing and recognizing text in lecture videos. In *2018 16th international conference on frontiers in handwriting recognition (ICFHR)*, pages 235–240. IEEE, 2018. 1, 2, 3
- [6] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 226–231, 1996. 2, 6, 8
- [7] Damianos Galanopoulos and Vasileios Mezaris. Temporal lecture video fragmentation using word embeddings. In *MultiMedia Modeling: 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8–11, 2019, Proceedings, Part II 25*, pages 254–265. Springer, 2019. 1, 2, 3
- [8] Zhangxuan Gu, Changhua Meng, Ke Wang, Jun Lan, Weiqiang Wang, Ming Gu, and Liqing Zhang. Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4583–4592, 2022. 6, 8
- [9] Jaekyu Ha, R.M. Haralick, and I.T. Phillips. Recursive x-y cut using bounding boxes of connected components. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, pages 952–955, 1995. 2, 6
- [10] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4083–4091, 2022. 1, 3, 5, 7
- [11] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*, pages 1–6, 2020. 7
- [12] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pages 498–517. Springer, 2022. 7
- [13] JP Kincaid. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. *Chief of Naval Technical Training*, pages 1–45, 1975. 4
- [14] Dong Won Lee, Chaitanya Ahuja, Paul Pu Liang, Sanika Natu, and Louis-Philippe Morency. Lecture presentations multimodal dataset: Towards understanding multimodality in educational videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20087–20098, 2023. 1, 2, 3, 4
- [15] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, 2020. Association for Computational Linguistics. 1, 3, 7
- [16] Chenliang Li, Haiyang Xu, Junfeng Tian, Wei Wang, Ming Yan, Bin Bi, Jiabo Ye, He Chen, Guohai Xu, Zheng Cao, Ji Zhang, Songfang Huang, Fei Huang, Jingren Zhou, and Luo Si. mPLUG: Effective and efficient vision-language learning by cross-modal skip-connections. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7241–7259, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics. 1, 3, 7
- [17] Irene Li, Alexander R Fabbri, Robert R Tung, and Dragomir R Radev. What should i learn first: Introducing lecturebank for nlp education and prerequisite chain learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6674–6681, 2019. 1, 2, 3
- [18] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 4
- [19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, pages 1–19, 2017. 7
- [20] T. Muthuprasad, S. Aiswarya, K.S. Aditya, and Girish K. Jha. Students’ perception and preference for online education in india during covid -19 pandemic. *Social Sciences Humanities Open*, 3(1):100101, 2021. 1
- [21] Pitambar Paudel. Online education: Benefits, challenges and strategies during and after covid-19 in higher education. *International Journal on Studies in Education (IJonSE)*, 3:70–85, 2021. 1

- [22] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020. 1, 3, 7
- [23] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992. Association for Computational Linguistics, 2019. 6
- [24] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987. 6
- [25] Anna Sun and Xiufang Chen. Online education and its effective practice: A research review. *Journal of information technology education: Research*, 15:1–34, 2016. 1
- [26] Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. Visualmrc: Machine reading comprehension on document images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13878–13888, 2021. 7
- [27] Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. Slidevqa: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13636–13645, 2023. 1, 2, 3, 4, 7
- [28] Zineng Tang, Ziyi Yang, Guoxin Wang, Yuwei Fang, Yang Liu, Chenguang Zhu, Michael Zeng, Cha Zhang, and Mohit Bansal. Unifying vision, text, and layout for universal document processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19254–19264, 2023. 1, 3, 5, 6, 7
- [29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 6000–6010, 2017. 6
- [30] Jiawei Wang, Kai Hu, Zhuoyao Zhong, Lei Sun, and Qiang Huo. Detect-order-construct: A tree construction based approach for hierarchical document structure analysis. *arXiv preprint arXiv:2401.11874*, pages 1–35, 2024. 5
- [31] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 3
- [32] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1192–1200, 2020. 1, 3
- [33] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. LayoutLMv2: Multi-modal pre-training for visually-rich document understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2579–2591, Online, 2021. Association for Computational Linguistics. 1, 3, 7
- [34] Antoine Yang, Arsha Nagraani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10714–10726, 2023. 1, 3, 7
- [35] Ori Yoran, Alon Talmor, and Jonathan Berant. Turning tables: Generating examples from semi-structured tables for endowing language models with reasoning skills. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 6016–6031. Association for Computational Linguistics, 2022. 7
- [36] Pengchuan Zhang, Xiujuan Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5579–5588, 2021. 3, 7