Weakly Supervised Learning for Textbook Question Answering

Jie Ma [1], Qi Chai [2], Jingyue Huang [2], Jun Liu [2], Yang You [2], and Qinghua Zheng [2]

[1]Xian Jiaotong University
[2]Affiliation not available

October 30, 2023

Abstract

Textbook Question Answering (TQA) is the task of answering diagram and non-diagram questions given large multi-modal contexts consisting of abundant text and diagrams. Deep text understandings and effective learning of diagram semantics are important for this task due to its specificity. In this paper, we propose a Weakly Supervised learning method for TQA (WSTQ), which regards the incompletely accurate results of essential intermediate procedures for this task as supervision to develop Text Matching (TM) and Relation Detection (RD) tasks and then employs the tasks to motivate itself to learn strong text comprehension and excellent diagram semantics respectively. Specifically, we apply the result of text retrieval to build positive as well as negative text pairs. In order to learn deep text understandings, we first pre-train the text understanding module of WSTQ on TM and then fine-tune it on TQA. We build positive as well as negative relation pairs by checking whether there is any overlap between the items/regions detected from diagrams using object detection. The RD task forces our method to learn the relationships between regions, which are crucial to express the diagram semantics. We train WSTQ on RD and TQA simultaneously, \emph{i.e.}, multitask learning, to obtain effective diagram semantics and then improve the TQA performance. Extensive experiments are carried out on CK12-QA and AI2D to verify the effectiveness of WSTQ. Experimental results show that our method achieves significant accuracy improvements of 5.02% and 4.12% on test splits of the above datasets respectively than the current state-of-the-art baseline. We have released our code on \url{https://github.com/dr-majie/WSTQ}.

Weakly Supervised Learning for Textbook Question Answering

Jie Ma, Qi Chai, Jingyue Huang, Jun Liu, *Senior Member, IEEE,* Yang You and Qinghua Zheng

***Abstract*—Textbook Question Answering (TQA) is the task of answering diagram and non-diagram questions given large multi-modal contexts consisting of abundant text and diagrams. Deep text understandings and effective learning of diagram semantics are important for this task due to its specificity. In this paper, we propose a Weakly Supervised learning method for TQA (WSTQ), which regards the incompletely accurate results of essential intermediate procedures for this task as supervision to develop Text Matching (TM) and Relation Detection (RD) tasks and then employs the tasks to motivate itself to learn strong text comprehension and excellent diagram semantics respectively. Specifically, we apply the result of text retrieval to build positive as well as negative text pairs. In order to learn deep text understandings, we first pre-train the text understanding module of WSTQ on TM and then fine-tune it on TQA. We build positive as well as negative relation pairs by checking whether there is any overlap between the items/regions detected from diagrams using object detection. The RD task forces our method to learn the relationships between regions, which are crucial to express the diagram semantics. We train WSTQ on RD and TQA simultaneously, *i.e.*, multitask learning, to obtain effective diagram semantics and then improve the TQA performance. Extensive experiments are carried out on CK12-QA and AI2D to verify the effectiveness of WSTQ. Experimental results show that our method achieves significant accuracy improvements of 5.02% and 4.12% on test splits of the above datasets respectively than the current state-of-the-art baseline. We have released our code on https://github.com/dr-majie/WSTQ.**

***Index Terms*—Textbook question answering, multi-modality, diagram understanding.**

I. Introduction

QUESTION answering, such as machine reading comprehension [1, 2] and visual question answering [3–5], has attracted extensive attention due to its popularity in some intriguing real-world applications, *e.g.*, autonomous driving [6] and image retrieval [7]. Recently, a new task Textbook Question Answering (TQA) [8, 9] possessing both of the characteristics of machine reading comprehension and visual question answering pushes forward vision-and-language comprehension. In particular, TQA is the task of answering diagram and non-diagram questions given multi-modal contexts shown in Figure 1, which is analogous to the real-life process of a human learning new knowledge from a lesson and estimating achievements. Taking a diagram question as an example, this task requires a system to have deep semantic understandings of multi-modal inputs and then predict answers accurately.

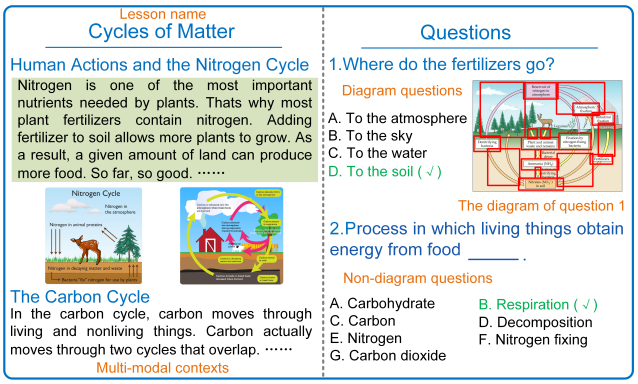


Fig. 1: An example of the TQA task. The olive background text (the diagram on the right) is the key textual (visual) knowledge to answer Question 1. The dataset contains no annotations other than the answer label. The red boxes within the right diagram are detected by YOLO [10].

TQA presents some challenges due to its specificity. First, it is difficult to learn deep semantic understandings of long textual contexts with limited training data, *e.g.*, only 15,153 samples in the CK12-QA[1] train split [9]. The textual contexts, especially the most relevant text of questions, are very important to predict answers. For example, the text with olive backgrounds in Figure 1 is the key knowledge to answer Question 1. Secondly, it is difficult to learn effective semantic representations of diagrams without annotations. The semantics of diagrams in textbooks, which are also very essential to predict answers, are expressed by a collection of items with 2-dimensional positions and a collection of relationships between items. Such relationships are expressed by the connections or overlaps between the items. For example, the diagram of Question 1 shown on the right of Figure 1 depicts nitrogen cycles by the overlaps between regions and contains the important visual knowledge to answer this question, *i.e.*, the fertilizer flow direction. However, there do not exist annotations for diagrams such as items and relationships.

In this paper, we propose a Weakly Supervised learning method for TQA (WSTQ), in which the incompletely ac-

Jie Ma and Jun Liu are with National Engineering Lab for Big Data Analytics, School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China (email: dr.majie@foxmail.com; liukeen@xjtu.edu.cn;).

Qi Chai and Qinghua Zheng are with Shaanxi Province Key Laboratory of Satellite and Terrestrial Network Tech. R&D, School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China (email: wyx1566@stu.xjtu.edu.cn; lijunjun_xd@outlook.com).

Jingyue Huang and Yang You are with the High Performance Computing for Artificial intelligence Lab, Department of Computer Science, National University of Singapore, Singapore 117417 (youy@comp.nus.edu.sg).

[1]The TQA dataset is collected from http://www.ck12.org. In this paper, we call the TQA dataset CK12-QA to distinguish TQA tasks from TQA datasets.

curate results of essential intermediate procedures for TQA are regarded as supervision to develop Text Matching (TM) and Relation Detection (RD) tasks, and then the above tasks motivate the system to learn strong text comprehension and excellent diagram semantics respectively. Concretely, we apply the result of text retrieval that is an important intermediate procedure for TQA [11, 12] and is used to find the most relevant text of questions, to develop TM. We consider the text t_i, which is most relevant to the question q_i in the lesson l_u, as the matching text of q_i and regard the text t_j of $q_j \in l_v, u \neq v$, as the mismatching text of q_i. The text understanding module of WSTQ is first pre-trained on TM and then fine-tuned on TQA to learn deep text understandings. We construct positive and negative relation pairs by checking whether there is any overlap between the items/regions detected from diagrams by object detection to develop RD. The detection is also an important procedure of TQA [13, 14] and is used to obtain diagram features. The RD task forces our method to learn the relationships between regions, which are crucial to express the diagram semantics. To learn effective diagram semantics and improve TQA performance, our method is trained on RD and TQA simultaneously, in which the parameters of the diagram understanding module are shared by both tasks, *i.e.*, multitask learning. It is worth noting that TM and RD are developed automatically rather than manually.

We evaluate WSTQ on two TQA datasets including CK12-QA [9] and AI2D [8]. Experimental results show that our method achieves the new State-Of-The-Art (SOTA) accuracy of 52.61% and 72.05% on CK12-QA and AI2D test splits respectively. To summarize, our contributions are mainly three-fold.

1) We propose a novel multitask learning framework that applies TM and RD to drive WSTQ to deepen the text understanding and learn the effective diagram semantics respectively.
2) We propose a weakly supervised developing strategy that uses the results of essential intermediate procedures for TQA to build TM and RD automatically.
3) We conduct experiments and ablation studies on CK12-QA and AI2D extensively to verify the effectiveness of WSTQ. We are the first to report the performance on various types of questions such as *what* and *how* within the mentioned datasets.

The remainder of this paper is organized as follows. Section II introduces the related works. Section III describes the task formulation. The details of our method is described in Section IV. The experiments on CK12-QA and AI2D are discussed in Section V. Finally, we make the concluding remarks in Section VI.

II. Related Work

Researchers have proposed various TQA methods, which try to address either multi-modality interaction or explainability challenges. In this section, we introduce how they address the issues.

Multi-modality Interaction The information interactions between questions and multi-modal contexts play a key role in predicting answers. Kembhavi *et al.* [8] first softly embedded textual contexts that are most relevant to questions as well as candidate answers via an attention mechanism and then projected textual and visual representations into a common space to predict answers. IGMN [15] finds the contradictions between textual contexts and candidate answers to build contradiction entity relationship graphs and then reasons over multi-modal inputs in the instructor of graphs. In contrast, F-GCN [12] applies graph convolutional networks [16] on textual contexts and diagrams to build unified graphs that memorize relevant question background information and predicts answers by reasoning over the graphs. EAMB [17] applies the essay-anchor attentive multi-modal bi-linear pooling method to learn the joint representations of text and diagrams. It first builds textual graphs based on textual contexts and then applies bilinear-based MFB [18] model to fuse graph and diagram representations. MoQA [19] regards textual contexts and diagrams as knowledge and then selects the top K most similar knowledge to answer questions. It also explores the TQA performance obtained by different information representations. All of the above methods were only conducted on the TQA validation split [9] due to the unavailable test split at that time and they were end-to-end trained only on CK12-QA. By comparison, ISAAQ [14] achieved SOTA results relying on fine-tuning large pre-trained models, ensemble learning and large datasets. The textual ISAAQ is pre-trained on RACE [20], ARC-Easy, ARC-Challenge [21] and OpenBookQA [22] datasets and fine-tuned on CK12-QA. Similarly, the multi-modal ISAAQ is pre-trained on VQA abstract scenes, VQA [3] and AI2D [8] datasets and fine-tuned on CK12-QA.

Explainability Practical TQA methods should not only answer textbook questions but provide students with explanations accurately, which helps them have a deeper understanding of what they have learned. There is only one work XTQA [11] researching on the TQA explainability. It regards the whole textual contexts of lessons as candidate evidence and applies a coarse-to-fine grained algorithm to extract span-level explanations for answering questions. However, it can only provide textual explanations for students rather than both textual and visual explanations.

The challenges WSTQ tries to address are different from the above works. Similar to RAFR [23], our method tries to learn effective diagram representations. However, RAFR only considers the text in diagrams, which causes the loss of visual information. By contrast, WSTQ not only considers the region representations but the relationships between them to learn more effective diagram semantics.

III. Task Formulation

The questions can be classified into three categories including Non-Diagram True or False (NDTF) with two candidate answers, Non-Diagram Multiple-Choice (NDMC) with four to seven candidate answers, and Diagram-Multiple-Choice (DMC) with four candidate answers. Following previous works [11, 14], we split TQA into NDTF, NDMC and DMC. We regard NDMC and DMC as a multi-class classification and consider NDTF as a binary classification. We only use the text

of multi-modal contexts due to the lack of diagrams in some lessons. An example of multi-modal contexts is shown on the left of Figure 1. In this section, we describe the formulation of each subtask.

NDTF and NDMC Given a dataset $\mathcal{S}_\psi$ consisting of N_ψ triples $(q_i, t_i, \mathcal{A}_i)$ with a question $q_i \in \mathcal{Q}_\psi$, text $t_i \in \mathcal{T}_\psi$ and candidate answers $\mathcal{A}_i \in \mathcal{A}_\psi$, NDTF and NDMC can be defined as follows:

$$\hat{a}_i = \underset{a_{i,m} \in \mathcal{A}_i}{\arg\max}\, p(a_{i,m}|q_i, t_i), \tag{1}$$

where $a_{i,m}$ denotes the m-th candidate answer for q_i and $\hat{a}_i$ denotes the predicted class. We use $|\mathcal{A}_i|$ denotes the number of candidate answers. For NDTF, $|\mathcal{A}_i| = 2$ and $|\mathcal{A}_i| = 7$ for NDMC.

DMC Given a dataset $\mathcal{S}_\phi$ consisting of N_ϕ quadruples $(q_k, d_k, t_k, \mathcal{A}_k)$ with a question $q_k \in \mathcal{Q}_\phi$, a diagram $d_k \in \mathcal{D}_\phi$, text $t_k \in \mathcal{T}_\phi$ and candidate answers $\mathcal{A}_k \in \mathcal{A}_\phi$, DMC can be defined as follows:

$$\hat{a}_k = \underset{a_{k,m} \in \mathcal{A}_k}{\arg\max}\, p(a_{k,m}|q_k, d_k, t_k), \tag{2}$$

where $a_{k,m}$ denotes the m-th candidate answer for q_k, $\hat{a}_k$ denotes the predicted class and $|\mathcal{A}_k| = 4$.

To describe the differences between subtasks, we use different subscripts such as q_i and q_k. In the following subsections, we do not use the subscripts to distinguish questions belonging to different subtasks, which may make readers easily understand our method. Following previous works [11, 14], we devise a corresponding method for answering the questions of each subtask respectively.

IV. Method

In this section, we first provide a brief overview of the architecture of our method. Then, we describe weakly supervised learning methods for NDTF and NDMC. Finally, we introduce a weakly supervised multitask learning method for DMC.

A. Overview

Figure 2 depicts the architecture of our method. We show the full forms of abbreviations in this figure on the bottom right. The TM developing is shown on the bottom left of Figure 2. Here, $(q_0 + \mathcal{A}_0, t_0)$ and $(q_0 + \mathcal{A}_0, t_5)$ are positive and negative TM pairs respectively, where $\mathcal{A}_0$ denotes the candidate answers of the question q_0, t_0 denotes the most relevant text extracted by information retrieval methods of q_0, and t_5 denotes the most relevant text of q_5.

We show the RD developing on the bottom right of Figure 2, where $r_{i,j}$ denotes the j-th region detected by object detection methods within diagram d_i. Region pairs with overlaps such as $(r_{i,15}, r_{i,17})$ are considered as positive samples. Instead, region pairs without overlaps such as $(r_{i,0}, r_{i,1})$ are regarded as negative samples. In non-diagram question answering, we first pre-train the text understanding module on TM and then fine-tune it on NDTF and NDMC respectively. In diagram question answering, the parameters of the diagram understanding module are shared by DMC and RD. The text understanding module pre-trained on TM is also fine-tuned on DMC. We train our method on DMC and RD jointly.

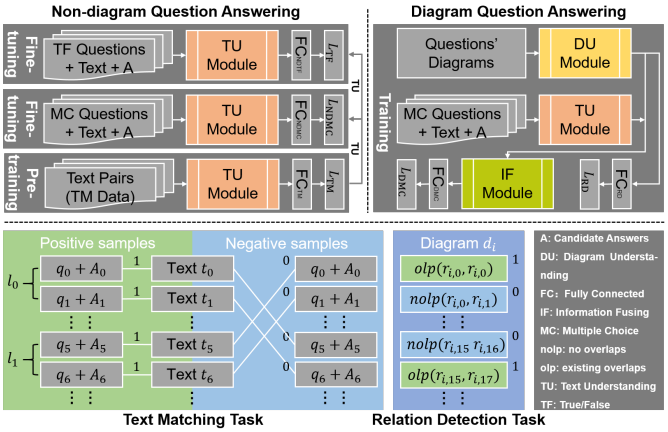


Fig. 2: The architecture of our method (WSTQ). We assume there are 10 questions and use them to develop TM in this illustration. The full forms of abbreviations are shown on the bottom right. The parameters of DU are shared with DMC and RD tasks.

B. Weakly Supervised Learning for NDTF and NDMC

The text understanding is important to answer questions accurately due to the TQA specificity but it may not be learned well using limited training data, *e.g.*, only 15,153 samples in the CK12-QA train split. Text retrieval is an essential intermediate procedure of TQA because it is used to find the most relevant text of questions. Inspired by this, we regard the incompletely accurate results of text retrieval methods as supervision to develop the TM task and train the text understanding module on TM to overcome the mentioned issue.

Information Retrieval (IR), Next Sentence Prediction (NSP), and Nearest Neighbors (NN) methods are applied to retrieve the most relevant text of questions respectively following the previous work [14]. Particularly, we first concatenate the question q_i and its candidate answers $\mathcal{A}_i$ as a query. Then, (1) a traditional search engine like ElasticSearch is used to perform IR. (2) we treat the text retrieval task as NSP using a Transformer [24] with frozen parameters. (3) we apply the Transformer to obtain the representations of queries and sentences within the textual context respectively and compute the cosine similarity between them to obtain NN. The text retrieval methods can also be replaced by other technologies such as TF-IDF [25]. WSTQ applies the above three methods respectively to explore their differences on TQA performance.

Relevant knowledge may exist in adjacent lessons due to the TQA specificity, *e.g.*, carbon and living things in Lesson 1 and carbon cycle in Lesson 2. This would cause a situation where negative text pairs are relevant. To address this issue, we devise a strategy to develop a relatively precise TM task. Specifically, we sort all the questions according to the lesson order and regard $(q_i + \mathcal{A}_i, t_i)$ as positive text pairs and $(q_i + \mathcal{A}_i, t_j)$ as negative text pairs, where $j = \big((N/2 + i) \mod N\big)$, $N =$

$N_\psi + N_\phi$ denotes the number of questions in the dataset, and t_j is the most relevant text of q_j.

Obviously, achieving high performance on TM and TQA requires a deep understanding of the text t_i. Inspired by this, we apply TM, which is developed automatically via weak supervision, to drive the text understanding module to learn deep text understandings. Specifically, we first train the text understanding module on TM by optimizing a binary cross-entropy loss function $\mathcal{L}_{\mathrm{TM}}$. Then, we fine-tune it on NDTF and NDMC by optimizing $\mathcal{L}_{\mathrm{NDTF}}$ and $\mathcal{L}_{\mathrm{NDMC}}$ respectively, which denote binary and multi-class cross-entropy loss functions. RoBERTa [26] is applied as the text understanding module to learn the joint representations of q_i, $a_{i,m}$ and t_i, and it can be replaced by existing text representing methods.

C. Weakly Supervised Multitask Learning for DMC

Object detection is also an important intermediate procedure of TQA and is used to extract image/diagram features [13, 14, 27]. The relationships expressed by the connections or overlaps between regions play a key role in expressing the semantics of diagrams. Inspired by these, we first apply object detection methods such as YOLO [10] to detect regions and check whether they have overlaps to develop positive as well as negative relation pairs. Then, we devise the multitask learning architecture to drive WSTQ to learn on not only RD but DMC tasks. This enables our method to learn effective semantic representations of diagrams and achieve good DMC performance.

Diagram Understanding (DU) WSTQ applies CNNs such as ResNet [28] to learn a x-dimensional vector of the k-th region $r_{i,k}$ detected by YOLO within the diagram d_i. The coordinate $c_{i,k} \in \mathbb{R}^4$ of $r_{i,k}$ is projected into a x-dimensional position vector using a Fully Connected (FC) layer due to its importance to relationship representations. Our method considers the arithmetic mean of them to be the representation $d_i^{'} \in \mathbb{R}^{\mu \times x}$ of d_i as follows:

$$d_i^{'} = \frac{\mathrm{LN}\big(\mathrm{CNNs}(r_i)\big) + \mathrm{LN}\big(c_i W_c\big)}{2}, \tag{3}$$

where μ denotes the number of regions within d_i, LN denotes the layer normalization [29] and $W_c \in \mathbb{R}^{4\times x}$ denotes the learned weight matrix.

RD Optimization Due to the lack of diagram annotations, our method only learns the implicit relations instead of explicit ones such as (*subject, relationship type, object*) in the visual relation detection task [30, 31]. To obtain the relationship scores between regions, our method first repeats the first and second dimension data of $d_i^{'}$ μ times, which is denoted as $d_i^{'0} \in \mathbb{R}^{\mu\times\mu\times x}$ and $d_i^{'1} \in \mathbb{R}^{\mu\times\mu\times x}$ respectively. Then, they are multiplied using the Hadamard product $\odot$ to obtain the joint representations $d_i^{''} \in \mathbb{R}^{\mu\times\mu\times x}$. Finally, WSTQ applies a FC layer to infer the relationship scores $s_i^r \in \mathbb{R}^{\mu\times\mu}$. The above steps can be denoted as follows:

$$\begin{aligned} d_i^{''} &= d_i^{'0} \odot d_i^{'1}, \\ s_i^r &= d_i^{''} W_r, \end{aligned} \tag{4}$$

where $W_r \in \mathbb{R}^x$ denotes the learned weight matrix.

Our method regards RD as a binary classification. In this task, negative relationship pairs are much more than positive pairs. For example, the diagram with 18 regions shown on the right of Figure 1 has 18*18=324 possible relationship pairs but only exists 51 positive relationship pairs. In order to make the positive samples being focused on, a weighted binary cross-entropy loss function $\mathcal{L}_{\mathrm{RD}}$ is applied as follows:

$$\begin{aligned} \mathcal{L}_{\mathrm{RD}} &= -\sum_{i=1}^{N_\phi} \Big(w_+ y_i^r \log \hat{y}_i^r + w_- (1 - y_i^r) \log(1 - \hat{y}_i^r) \Big), \\ \hat{y}_i^r &= \sigma(s_i^r), \end{aligned} \tag{5}$$

where N_ϕ denotes the number of questions within DMC, w_+ and w_- denote the weights of positive and negative relationship pairs respectively, $y_i^r \in \{0,1\}^{\mu^2}$ denotes the labels of relationship pairs within d_i, $\hat{y}_i^r \in [0,1]^{\mu^2}$ denotes the probability of relationship pairs being predicted as positive classes, μ^2 denotes the number of possible relationship pairs within d_i and σ denotes the sigmoid function.

Text Understanding (TU) Learning deep understandings of $\mathbf{t}_i$ is also important for answering questions of DMC. Hence, the text understanding module pre-trained on TM is applied to learn text representations, which has the same setting as the above subsection. For simplicity, WSTQ applies this module to learn the joint representations $e_{i,m} \in \mathbb{R}^x$ of t_i, q_i and $a_{i,m}$ as follows:

$$e_{i,m} = \mathrm{TU}(t_i, q_i, a_{i,m}), \tag{6}$$

where TU is RoBERTa.

Information Fusing Attention mechanisms are widely used to obtain the attended representations of diagrams. For example, top-down [13] and question-guided attention mechanisms [23] are used to learn the global attended image representations, which can improve the performance. However, the attention mechanisms cause reductions of TQA performance in WSTQ. Therefore, our method treats the weight of each region as the same and obtains the global diagram representations $d_i^\alpha \in \mathbb{R}^x$ by summing the representation of each region. To obtain the multi-modal fusion representations $f_i \in \mathbb{R}^{|\mathcal{A}_i|\times x}$ of $e_i \in \mathbb{R}^{|\mathcal{A}_i|\times x}$ and $d_i^\alpha \in \mathbb{R}^x$, WSTQ applies the Hadamard product $\odot$ to fuse them. The mentioned steps can be denoted as follows:

$$\begin{aligned} d_i^\alpha &= \sum_{i=1}^{\mu} d_i^{'}, \\ f_i &= e_i \odot d_i^\alpha, \end{aligned} \tag{7}$$

where $d_i^{'}$ is the learned diagram representations with μ regions and $|\mathcal{A}_i|$ denotes the number of candidate answers of q_i.

DMC Optimization WSTQ uses a FC layer to predict the scores of candidate answers $\mathbf{s}_i^a \in \mathbb{R}^{|\mathbf{A}_i|}$ as follows:

$$s_i^a = f_i W_a, \tag{8}$$

where $W_a \in \mathbb{R}^x$ denotes the learned weight matrix. WSTQ regards DMC as a multi-class classification and applies the

multi-class cross-entropy loss function $\mathcal{L}_{\mathrm{DMC}}$ to optimize answer predicting as follows:

$$\begin{aligned} \mathcal{L}_{\mathrm{DMC}} &= -\sum_{i=1}^{N_\phi} y_i^a \log \hat{y}_i^a, \\ \hat{y}_i^a &= \mathrm{softmax}(s_i^a), \end{aligned} \tag{9}$$

where $y_i^a \in \{0,1\}^{|\mathcal{A}_i|}$ denotes the answer label, $\hat{y}_i^a \in [0,1]^{|\mathcal{A}_i|}$ denotes the probability of candidate answers belonging to their corresponding classes and $\mathrm{softmax}$ denotes the softmax function.

Multitask Optimization To optimize DMC as well as RD simultaneously, the weighted sum $\mathcal{L}_{\mathrm{MTL}}$ of $\mathcal{L}_{\mathrm{DMC}}$ and $\mathcal{L}_{\mathrm{RD}}$ is applied as follows:

$$\mathcal{L}_{\mathrm{MTL}} = \mathcal{L}_{\mathrm{DMC}} + \lambda \mathcal{L}_{\mathrm{RD}}, \tag{10}$$

where λ denotes the weight to adjust $\mathcal{L}_{\mathrm{RD}}$.

V. Experiments

In this section, we first describe the experimental setups such as evaluation datasets and implementation details. Then, the results of each subtask within CK12-QA and AI2D are discussed. Third, we introduce the ablation studies on CK12-QA. Finally, we show the results on various type of questions such as *how* and *what*.

A. Experimental Setup

Datasets and Evaluation Metrics To the best of our knowledge, existing TQA methods except ISAAQ [14] are only evaluated on CK12-QA. Following [14], we evaluate WSTQ not only on CK12-QA [9] but AI2D [8], which contains textbook (diagram) questions. Specifically, CK12-QA is developed from middle school curricula including life science, earth science and physical science. It is split into a training split with 666 lessons, a validation split with 200 lessons and a test split with 210 lessons. AI2D is developed from grade school curricula and only consists of diagram questions. The detailed statistic on each split of the mentioned datasets is shown in Table I. In CK12-QA, NDTF, NDMC and DMC have 3,490, 5,162 and 6,501 training questions respectively. In AI2D, DMC contains 7,824 training questions. Please Note AI2D only contains DMC questions. Following previous works [9, 11, 14], we use accuracy to evaluate our method.

TABLE I: The number of questions within each subtask of CK12-QA [8] and AI2D [8]. AI2D does not have NDTF and NDMC questions.

Subtask	CK12-QA			AI2D		
	train	validation	test	train	validation	test
NDTF	3,490	998	912	-	-	-
NDMC	5,162	1,530	1,600	-	-	-
DMC	6,501	2,781	3,285	7,824	906	978

Implementation Details We introduce the implementation detail of each module within WSTQ as follows. In DU, the pre-trained ResNet-101 backbone is fine-tuned to learn the $x = 1024$ dimensional representation of each region within diagrams. In RD Optimization, WSTQ applies YOLO that is fine-tuned on AI2D [8] with an initial learning rate $1e^{-4}$ to detect regions within diagrams. Our method applies $w_+ = 1.5$ and $w_- = 1$ to optimize the relation detection. The RoBERTa-large [26] is applied to be the text understanding module, which is first fine-tuned on TM and then fine-tuned on subtasks of CK12-QA. Our method selects the maximum input sequences of 64 tokens for NDTF and 180 for NDMC, DMC and TM. In Multitask Optimization, $\lambda = 0.1$ is used to be the weight of $\mathcal{L}_{\mathrm{RD}}$.

Our method is trained during 6 epochs by Adam optimizer [32] with linearly-decayed learning rate and warm-up. We select the initial learning rate $1e^{-5}$ for NDTF, $2.5e^{-6}$ for NDMC, DMC and $1e^{-6}$ for TM. The dropout value 0.1 is chosen to avoid over-fitting. We implement WSTQ based on PyTorch and run our code on one NVIDIA Tesla V100 card.

B. Results on CK12-QA

Comparison with SOTA Baselines We compare WSTQ with the previous SOTA methods on CK12-QA validation and test splits. We select XTQA [11], RAFR [23], ISAAQ [14] to be baselines because the other works introduced in Section II lack the results on the test split. The authors of these works have not released their codes. RAFR analyzes the dependencies between text within diagrams to build visual graphs and then applies dual attentions to predict answers. It obtains the best performance on validation splits compared with the model without pre-training and fine-tuning. XTQA achieves the best results on the test splits under the mentioned comparison conditions. However, their results are rather modest. ISAAQ achieves the current SOTA results based on fine-tuning the large pre-trained model, training on large datasets and ensemble learning. Please see details in Section II.

We select three ISAAQ versions including $\mathrm{ISAAQ_{IR}}$, $\mathrm{ISAAQ_{NSP}}$, and $\mathrm{ISAAQ_{NN}}$ that are trained only on CK12-QA to fairly compare with our method. Please see details about IR, NSP and NN in Section IV-B. We run the codes of ISAAQ and WSTQ three times on the same machine with random seeds. The best result of each time is selected to compute the average and standard deviation.

Results Table II shows the main result on CK12-QA validation and test splits. We can see that $\mathrm{WSTQ_{IR}}$ significantly outperforms the current SOTA method $\mathrm{ISAAQ_{IR}}$, improving the accuracy on the whole questions of the test split from 47.78% to 52.61%. The improvement is observed consistently on other versions of WSTQ, *e.g.*, $\mathrm{WSTQ_{NN}}$ outperforms $\mathrm{ISAAQ_{NN}}$ by 5.02% on all the questions of the test split. Our method performs best on all subtasks, especially on DMC. It can be seen that WSTQ/ISAAQ with different text retrieval methods have significantly different performance, which demonstrates the importance of the text most relevant with questions. The traditional IR methods such as ElasticSearch may have the best retrieval performance. We will investigate how to retrieve the more accurate text in the future, which may improve the TQA performance substantially. We can also see the generalization

TABLE II: Accuracy (%) and significance test on CK12-QA validation and test splits of each subtask. NDALL denotes the accuracy on non-diagram questions. ALL denotes the accuracy on the whole questions within CK12-QA. NDALL=NDTF ∪ NDMC. ALL=NDALL ∪ DMC. Δ_* denotes the improvement of WSTQ$_*$ over ISAAQ$_*$. p-value$_*$ denotes the significance test (paired t-test) between WSTQ$_*$ and ISAAQ$_*$.

	Validation					**Test**				
	NDTF	NDMC	NDALL	DMC	ALL	NDTF	NDMC	NDALL	DMC	ALL
RAFR	53.63	36.67	43.35	32.85	37.85	52.75	34.38	41.03	30.47	35.04
XTQA	58.24	30.33	41.32	32.05	36.46	56.22	33.40	41.67	33.34	36.95
ISAAQ$_{\text{NN}}$	71.67±0.59	54.34±0.47	61.18±0.14	36.65±1.19	48.33±0.60	72.30±0.94	56.75±0.76	62.40±0.58	33.28±1.66	45.90±0.75
ISAAQ$_{\text{NSP}}$	72.01±2.15	54.45±0.57	61.38±1.04	39.54±0.76	49.94±0.38	71.71±1.16	55.31±1.33	61.26±1.27	35.51±1.28	46.67±0.20
ISAAQ$_{\text{IR}}$	74.32±1.46	55.53±0.15	62.95±0.66	41.20±0.70	51.55±0.05	75.29±1.17	56.33±0.07	63.21±0.38	35.98±0.76	47.78±0.40
WSTQ$_{\text{NN}}$	73.95±0.30	**57.89±2.61**	64.23±1.47	46.64±0.79	55.01±1.05	74.41±0.92	60.36±1.57	**65.46±0.67**	39.80±1.02	50.92±0.57
WSTQ$_{\text{NSP}}$	72.38±0.29	57.67±0.59	63.48±0.46	48.08±1.03	55.41±0.49	72.74±0.46	58.65±0.40	63.77±0.36	40.97±0.82	50.85±0.36
WSTQ$_{\text{IR}}$	**76.65 ± 0.62**	56.30±0.49	**64.33±0.54**	**50.04±0.78**	**56.85±0.32**	**76.68±0.60**	**57.98±0.29**	64.77±0.38	**43.32±0.96**	**52.61±0.63**
Δ_{NN}	+2.28	**+3.55**	**+3.05**	**+9.99**	**+6.68**	**+2.11**	**+3.61**	**+3.07**	+6.52	**+5.02**
Δ_{NSP}	+0.37	+3.22	+2.09	+8.54	+5.47	+1.03	+3.34	+2.50	+5.46	+4.18
Δ_{IR}	**+2.33**	+0.77	+1.39	+8.84	+5.30	+1.39	+1.65	+1.56	**+7.34**	+4.83
p-value$_{\text{NN}}$	$6.37e^{-2}$	$5.99e^{-2}$	$\mathbf{4.87e^{-2}}$	$\mathbf{1.28e^{-4}}$	$\mathbf{9.22e^{-6}}$	$1.41e^{-1}$	$\mathbf{6.64e^{-4}}$	$\mathbf{7.34e^{-3}}$	$\mathbf{4.86e^{-4}}$	$\mathbf{3.61e^{-4}}$
p-value$_{\text{NSP}}$	$7.82e^{-1}$	$\mathbf{2.45e^{-3}}$	$\mathbf{3.29e^{-2}}$	$\mathbf{9.74e^{-4}}$	$\mathbf{1.07e^{-4}}$	$2.26e^{-1}$	$\mathbf{1.41e^{-2}}$	$\mathbf{3.01e^{-2}}$	$\mathbf{3.40e^{-3}}$	$\mathbf{6.15e^{-5}}$
p-value$_{\text{IR}}$	$\mathbf{3.96e^{-3}}$	$8.13e^{-2}$	$\mathbf{2.32e^{-2}}$	$\mathbf{2.66e^{-4}}$	$\mathbf{6.67e^{-4}}$	$\mathbf{4.99e^{-2}}$	$\mathbf{2.31e^{-2}}$	$\mathbf{3.93e^{-3}}$	$\mathbf{4.41e^{-3}}$	$\mathbf{7.66e^{-4}}$

TABLE III: Accuracy (%) and significance test on the AI2D validation and test splits. Δ denotes the improvement of WSTQ over ISAAQ. p-value denotes the significance test (paired t-test) between WSTQ and ISAAQ.

Model	**Validation** DMC	**Test** DMC
DQA-NET	-	38.47
ISAAQ	68.88±1.70	67.93±0.53
WSTQ	**73.62±0.40**	**72.05±1.55**
Δ	**+4.74**	**+4.12**
p-value	$\mathbf{9.30e^{-3}}$	$\mathbf{1.21e^{-2}}$

ability of WSTQ and ISAAQ on DMC is slightly weaker than that on NDTF and NDMC, which may be caused by the difficulty of diagram understanding and the different data distribution between splits. For the former, explicit relations between regions like visual relation detection [30, 31, 33] may improve the diagram understanding. For the latter, fine grained attentions may enhance the reasoning ability to overcome the data shift [34, 35].

Furthermore, we conduct the pair-wise significance test (paired t-test) between WSTQ$_*$ and ISAAQ$_*$ on each subtask. We can see that WSTQ is significantly better than ISAAQ except on NDTF ($p \leq 0.05$) within the test split. This demonstrates the effectiveness of our method. We can also see that the results of pre-training (fine-tuning) based methods such as our method and ISAAQ are better than RAFR and XTQA that are trained from scratch and do not use pre-training and fine-tuning. We can conclude that large pre-training models can bring a significant improvement on specific tasks with limited data.

C. Results on AI2D

Comparison with SOTA Baselines We also compare WSTQ with the previous SOTA methods on AI2D [8]. There have a few works on TQA and most of them only conducts experiments on CK12-QA. DQA-NET [8], which is the first work on this dataset, answers the questions within DMC by reasoning over the diagram parse graphs. It does not depend on pre-training and fine-tuning and its results are rather modest. ISAAQ [14] is the current best-performing method on this dataset as well. We choose ISAAQ trained only on AI2D to compare with our method fairly. AI2D contains no textual context and does not require to perform information retrieving. Therefore, there is only one version for our method and ISAAQ. We use the settings on CK12-QA to obtain the result on AI2D.

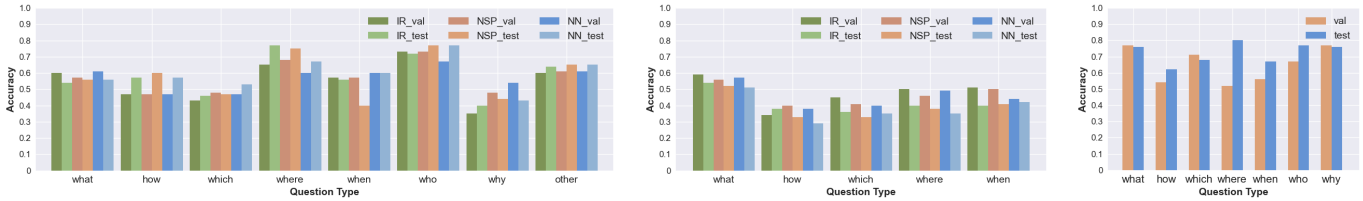
Results Table III shows the accuracy on validation and test splits of AI2D. We can see WSTQ achieves the new SOTA performance, significantly improving the accuracy on the test split by 4.12%. We also conduct the pair-wise significance test (paired t-test) between WSTQ and ISAAQ. It can be seen that our method is significantly better than ISAAQ on AI2D validation and test splits ($p \leq 0.05$). The results also show that our method and ISAAQ significantly outperforms DQA-NET. In summary, WSTQ pushes forward the SOTA results on two public datasets, demonstrating its effectiveness.

D. Ablation Study

In order to further analyze WSTQ, we carry out ablation studies shown in Table IV on the CK12-QA validation split. The performance differences obtained by different information retrieving methods have been shown in the previous subsections. Here, we only choose WSTQ$_{\text{IR}}$ to verify the effectiveness of each module.

W/O Diagrams We remove the diagrams to explore how well WSTQ performs on DMC. There do not exist diagrams in the questions of NDTF and NDMC. We can see that the DMC accuracy decreases by 2.05%, demonstrating the importance of diagrams for diagram question answering.

W/O TM We do not pre-train text understanding on TM to verify its effectiveness on text understanding. The accuracy of NDTF, NDMC, and DMC decreases by 4.40%, 2.46% and 1.12% respectively, demonstrating the importance of TM



(a) NDMC-CK12-QA (b) DMC-CK12-QA (c) DMC-AI2D

Fig. 3: Accuracy of WSTQ on various types of questions. We show the results on validation and test splits respectively. We classify the questions into 8 categories: *what*, *how*, *which*, *where*, *when*, *who*, *why* and *other*. Figure (3a) and Figure (3b) depict the detailed NDMC and DMC accuracy within CK12-QA respectively. Figure (3c) depicts the detailed DMC accuracy within AI2D.

TABLE IV: Ablation Study on the CK12-QA validation split. Δ denotes the accuracy decrease (%) without the specified module. DU represents the diagram understanding module. TM denotes the text matching task. RD denotes the relation detection task.

Model	NDTF	Δ	NDMC	Δ	DMC	Δ
WSTQ_{IR}	**77.15**		**56.80**		**50.92**	
W/O Diagrams	-	-	-	-	48.87	**-2.05**
W/O TM	72.75	**-4.40**	55.60	**-2.46**	49.80	-1.12
W/O RD	-	-	-	-	49.08	-1.84
Freezing DU	-	-	-	-	50.20	-0.72

on text understanding. The text understanding module is pre-trained on a binary classification TM task. It may match better on NDTF and make bigger contributions compared with that on NDMC because the former is also a binary classification task. Actually, we can see that the decrease on NDTF is more than that on NDMC. This setting also demonstrates the effectiveness of regarding information retrieval results as supervision to develop TM.

W/O RD The DMC accuracy drops from 50.92% to 49.08% and is close to the performance achieved by WSTQ w/o diagrams, which demonstrates the effectiveness of RD on semantic representations of diagrams and the effectiveness of regarding objection detection results as supervision to develop RD. Combined with the first and this setting, we can conclude that current diagram understanding has potential improvement. For example, learning explicit relations and building diagram graphs [36] under different relations may be effective.

Freezing DU We freeze the diagram understanding module to explore whether the RD loss in Eq. (5) can be used to optimize other modules. This loss is specifically designed to optimize the diagram understanding module, allowing WSTQ to learn effective semantic representations of diagrams. In this setting, the variant of our method performs RD with freezing DU, which means the RD loss is forced to optimize other modules. We can see that this variant of WSTQ outperforms that without RD by 1.12%, which proves our argument.

E. Results on Various Types of Questions

Previous works [12, 23] only reported the experimental results on NDTF, NDMC and DMC, making their model analyses less detailed. We classify the questions of DMC and NDMC into 8 categories like [37] to further analyze our model: *what*, *how*, *which*, *where*, *when*, *who*, *why* and *other*. The questions are classified by determining whether there exist the above-mentioned class labels. We think the results on these various types of questions can provide a comprehensive analysis for WSTQ. For example, achieving high performance on *when*, *why*, *where* and *how* questions usually require the high-level reasoning ability [37]. We do not classify the questions of NDTF because they do not contain the above-mentioned labels.

We show the experimental results in Figure 3. Concretely, Figure (3a) and (3b) show the detailed NDMC and DMC accuracy within CK12-QA respectively. The DMC subtask does not contain *when*, *who* and *why* questions. It can be seen that WSTQ obtains the better performance on all types of questions within NDMC compared with that within DMC, which demonstrates multi-modal question answering is more challenging than textual question answering. We can see the accuracy of WSTQ on *how* questions is not as good as that *what* questions within CK12-QA, because *how* questions may necessitate the high-level multi-hop reasoning ability. For *which* questions such as "which letter denotes the mitochondrion in this diagram", they usually require models to have a deep semantic understanding and word-region alignment.

Figure (3c) shows the detailed DMC accuracy within AI2D. We can see that WSTQ achieves good results on all types of questions within the test split, demonstrating its strong generalization and reasoning ability. Moreover, it can be seen that our method obtains the better accuracy on AI2D compared with that on CK12-QA. This may be caused by the following reasons: (1) Achieving high performance on questions within CK12-QA usually requires external knowledge but it may not be necessitated for questions within AI2D. (2) TQA and AI2D are developed from middle and grade school curricula respectively, which means questions within the former are more challenging than that of the latter. The accuracy on *how* questions is slightly worse than other types of questions, which shows the similar situation on CK12-QA.

F. Case Study

We present the case studies for our method. Figure 4 depicts a qualitative case for each TQA subtask respectively in order

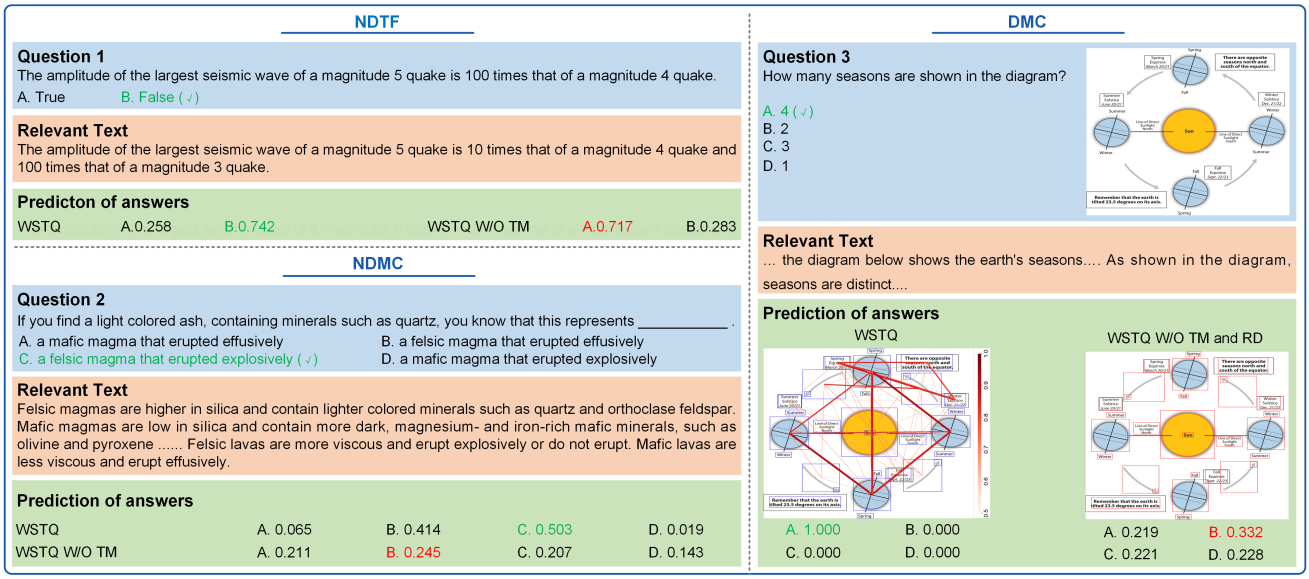


Fig. 4: Case studies for WSTQ_{IR} and its variant without pre-training the text understanding module on TM and without training the diagram understanding module on RD. The NDTF, NDMC and DMC examples are from CK12-QA. Our method and its variant use the same relevant text to answer the specific question. Answers marked in green are ground truth. Answers marked in red are wrong predictions.

to show the strengths of WSTQ_{IR} intuitively. WSTQ_{IR} and its variant without pre-training the text understanding module on TM and training the diagram understanding module on RD use the same relevant text to answer the specific question but make very different predictions.

NDTF As can be seen on the top left of Figure 4, the variant of WSTQ_{IR} may not fully comprehend the textual context and may make predictions solely based on text similarities. However, our method predicts the answer *B* with a high degree of certainty, demonstrating that TM motivates WSTQ_{IR} to learn a deeper text understanding.

NDMC On the bottom left of Figure 4, it can be seen that the variant predicts a close probability for each candidate answer, which shows that it may make predictions based on the text similarity. By contrast, our method predicts extremely different probabilities for (*A, D*) and (*B, C*). Moreover, WSTQ_{IR} can still predict the answer *C* accurately, although there has a long distance between *felsic lavas* and *explosively*. These demonstrate TM drives our method to have a deep understanding and summarization ability for long text. The above cases also show the strength of considering the results of information retrieval as supervision to develop TM in spite of some noise existing.

DMC The variant of WSTQ_{IR} without pre-training the text understanding module on TM and training the diagram understanding module on RD only considers the separate region information to be the representations of diagrams, resulting in incorrect predictions. Nevertheless, our method explicitly predicts the relationships between regions such as *seasons* for deep understandings of diagram semantics, making completely accurate predictions for the questions within DMC. This show the strength of regarding the results of object detection as supervision to develop RD despite some noise existing. In addition, this also intuitively shows RD and TQA can enhance each other, *i.e.*, the advantage of multitask learning.

In summary, even though the results of intermediate procedures contain some noisy supervision information (weak supervision), they can still motivate our method to learn deep understandings of long text and abstract diagrams with limited training data.

VI. Conclusion and Future Work

In this paper, we propose a weakly supervised learning method for TQA called WSTQ, which considers the intermediate procedures that are essential for this task as supervision to develop TM and RD tasks, and then uses them to drive itself to learn deep semantic understandings of text and diagrams respectively. To be more specific, the TM task motivates WSTQ to learn a deep text understanding. The RD task drives our method to take into account the relationships between regions, which are important in expressing diagram semantics. Extensive experiments and ablation studies demonstrate the effectiveness of WSTQ and the contribution of each module. We also show the experimental result on various types of questions, such as *where* and *when* to further analyze our method.

In the future, we will investigate the following directions.

1) We will explore how to generate the textual attribute for each detected region and devise an attribute-word guided attention mechanism to learn more effective vision-language representations.
2) We will explore how to obtain more accurate relevant textual context that is important to answer questions accurately.
3) We will explore how to detect the explicit relationships between regions and apply graph neural networks to learn diagram representations under specific relations.

ACKNOWLEDGMENTS

This work was supported by National Key Research and Development Program of China (2020AAA0108800), National Natural Science Foundation of China (61877050, 61937001, and 62020194), Innovative Research Group of the National Natural Science Foundation of China (61721002), Innovation Research Team of Ministry of Education (IRT_17R86), Project of China Knowledge Centre for Engineering Science and Technology, MoE-CMCC "Artifical Intelligence" Project (MCM20190701), The National Social Science Fund of China (18XXW005), National Statistical Science Research Project (2020LY103), Ministry of Education Humanities and Social Sciences Fund (17YJA860028), China Scholarship Council (202006280344) and the Natural Science Basic Research Plan in Shaanxi Province of China (2020JM-070).

REFERENCES

[1] M. Richardson, C. J. Burges, and E. Renshaw, "Mctest: A challenge dataset for the open-domain machine comprehension of text," in *Proceedings of the conference on empirical methods in natural language processing*, 2013, pp. 193–203.

[2] S. Sugawara, P. Stenetorp, K. Inui, and A. Aizawa, "Assessing the benchmarking capacity of machine reading comprehension datasets," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 8918–8927.

[3] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2425–2433.

[4] D. A. Hudson and C. D. Manning, "Gqa: A new dataset for real-world visual reasoning and compositional question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6700–6709.

[5] X. Chen, M. Jiang, and Q. Zhao, "Predicting human scanpaths in visual question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 876–10 885.

[6] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt *et al.*, "Towards fully autonomous driving: Systems and algorithms," in *2011 IEEE intelligent vehicles symposium (IV)*. IEEE, 2011, pp. 163–168.

[7] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys (Csur)*, vol. 40, no. 2, pp. 1–60, 2008.

[8] A. Kembhavi, M. Salvato, E. Kolve, M. Seo, H. Hajishirzi, and A. Farhadi, "A diagram is worth a dozen images," in *European Conference on Computer Vision*, 2016, pp. 235–251.

[9] A. Kembhavi, M. Seo, D. Schwenk, J. Choi, A. Farhadi, and H. Hajishirzi, "Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension," in *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, 2017, pp. 4999–5007.

[10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[11] J. Ma, J. Liu, J. Li, Q. Zheng, Q. Yin, J. Zhou, and Y. Huang, "Xtqa: Span-level explanations of the textbook question answering," *arXiv preprint arXiv:2011.12662*, 2020.

[12] D. Kim, S. Kim, and N. Kwak, "Textbook question answering with multi-modal context graph understanding and self-supervised open-set comprehension," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 3568–3584.

[13] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6077–6086.

[14] J. M. Gómez-Pérez and R. Ortega, "Isaaq-mastering textbook questions with pre-trained transformers and bottom-up and top-down attention," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2020, pp. 5469–5479.

[15] J. Li, H. Su, J. Zhu, S. Wang, and B. Zhang, "Textbook question answering under instructor guidance with memory networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3655–3663.

[16] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *5th International Conference on Learning Representations*. OpenReview.net, 2017.

[17] J. Li, H. Su, J. Zhu, and B. Zhang, "Essay-anchor attentive multi-modal bilinear pooling for textbook question answering," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2018, pp. 1–6.

[18] Z. Yu, J. Yu, J. Fan, and D. Tao, "Multi-modal factorized bilinear pooling with co-attention learning for visual question answering," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1821–1830.

[19] M. Haurilet, Z. Al-Halah, and R. Stiefelhagen, "Moqa - A multi-modal question answering architecture," in *Computer Vision - ECCV 2018 Workshops - Munich, Germany, September 8-14, 2018, Proceedings, Part IV*, ser. Lecture Notes in Computer Science, vol. 11132. Springer, 2018, pp. 106–113.

[20] G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy, "Race: Large-scale reading comprehension dataset from examinations," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 785–794.

[21] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord, "Think you have solved question answering? try arc, the ai2 reasoning challenge," *arXiv preprint arXiv:1803.05457*, 2018.

[22] T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal, "Can a suit of armor conduct electricity? a new dataset for open book question answering," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 2381–2391.
[23] J. Ma, J. Liu, Y. Wang, J. Li, and T. Liu, "Relation-aware fine-grained reasoning network for textbook question answering," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2021.
[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
[25] A. Chaturvedi, O. A. Pandit, and U. Garain, "Cnn for text-based multiple choice question answering," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 272–277.
[26] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.
[27] J. Ma, J. Liu, Q. Lin, B. Wu, Y. Wang, and Y. You, "Multitask learning for visual question answering," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.
[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[29] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.
[30] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua, "Visual translation embedding network for visual relation detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5532–5540.
[31] R. Wu, K. Xu, C. Liu, N. Zhuang, and Y. Mu, "Localize, assemble, and predicate: Contextual object proposal embedding for visual relation detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12 297–12 304.
[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations*, 2015.
[33] M. Diomataris, N. Gkanatsios, V. Pitsikalis, and P. Maragos, "Grounding consistency: Distilling spatial common sense for precise visual relationship detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 911–15 920.
[34] P. Oza, H. V. Nguyen, and V. M. Patel, "Multiple class novelty detection under data distribution shift," in *European Conference on Computer Vision*. Springer, 2020, pp. 432–449.
[35] S. Sankaranarayanan, Y. Balaji, A. Jain, S. N. Lim, and R. Chellappa, "Learning from synthetic data: Addressing domain shift for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3752–3761.
[36] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *European semantic web conference*. Springer, 2018, pp. 593–607.
[37] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, "Visual7w: Grounded question answering in images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4995–5004.

Jie Ma received his B.Eng degree from Lanzhou University of Technology, Lanzhou, China, in 2015, and M.Eng degree from Chongqing University, Chongqing, China, in 2018. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology at Xi'an Jiaotong University, Xi'an, China. He is currently co-supervised by Professor Yang You at the National University of Singapore. His research interests include textbook question answering, visual question answering, coreference resolution and imbalanced data classification.

Qi Chai received his B.E. degree from Xi'an Jiaotong University, Xi'an, China, in 2021. He is currently pursuing the M.E. degree with the School of Computer Science and Technology at Xi'an Jiaotong University, Xi'an, China. His research interests include multi-modal question answering ,image processing and natural language processing.

Jingyue Huang is a senior student at Fudan University studying Data Science. She was awarded an outstanding student scholarship in Fudan University and involved in a research project in concept extraction in the group of Prof. Deqing Yang since autumn 2020. She interned at HPC-AI Lab of National University of Singapore in summer 2021 under the supervision of Prof. Yang You. Her research interests include natural language processing and multi-modal learning. She is looking to pursue graduate school in the near future.

Jun Liu received the B.S. in computer science and technology in 1995 and Ph.D. degrees in systems engineering in 2004, both from Xi'an Jiaotong University, China. He is currently a Professor with the Department of Computer Science, Xi'an Jiaotong University. He has authored more than ninety research papers in various journals and conference proceedings. He has won the best paper awards in IEEE ISSRE 2016 and IEEE ICBK 2016. His research interests include NLP and e-learning. Dr. Liu currently serves as an associate editor of IEEE TNNLS from 2020, and has served as a guest editor for many technical journals, such as Information Fusion, IEEE SYSTEMS JOURNAL. He also acted as a conference/workshop/track chair at numerous conferences.

Yang You is a Presidential Young Professor at National University of Singapore. He is on an early career track at NUS for exceptional young academic talents with great potential to excel. He received his PhD in Computer Science from UC Berkeley. His advisor is Prof. James Demmel, who was the former chair of the Computer Science Division and EECS Department. Yang You's research interests include Parallel/Distributed Algorithms, High Performance Computing, and Machine Learning. The focus of his current research is scaling up deep neural networks training on distributed systems or supercomputers. In 2017, his team broke the world record of ImageNet training speed, which was covered by the technology media like NSF, ScienceDaily, Science NewsLine, and i-programmer. In 2019, his team broke the world record of BERT training speed. The BERT training techniques have been used by many tech giants like Google, Microsoft, and NVIDIA. Yang You's LARS and LAMB optimizers are available in industry benchmark MLPerf. He is a winner of IPDPS 2015 Best Paper Award (0.8%), ICPP 2018 Best Paper Award (0.3%) and ACM/IEEE George Michael HPC Fellowship. Yang You is a Siebel Scholar and a winner of Lotfi A. Zadeh Prize. Yang You was nominated by UC Berkeley for ACM Doctoral Dissertation Award (2 out of 81 Berkeley EECS PhD students graduated in 2020). He also made Forbes 30 Under 30 Asia list (2021) for young leaders. For more information, please check his lab's homepage at https://ai.comp.nus.edu.sg/.

Qinghua Zheng received the B.S. degree in computer software, the M.S. degree in computer organization and architecture, and the Ph.D. degree in system engineering from Xi'an Jiaotong University, Xi'an, China, in 1990, 1993, and 1997, respectively. He was a Postdoctoral Researcher with the Harvard University, Cambridge, MA, USA, in 2002.He is currently a Professor with the Department of Computer Science and Technology, Xi'an Jiaotong University. His research interests include knowledge engineering of big data and software trustworthiness evaluation.