HotpotNet for TextVQA

Hao Wu* **Jiayi Shen*** **Yanlin Feng*** **Yinghuan Zhang*** **Yuwei Wu***
{haowu3, jiayis2, yanlinf, yinghuan, yuweiwu}@andrew.cmu.edu

Abstract

In this project, we introduce HotpotNet with two pre-training tasks to improve the TextVQA model's robustness to the off-the-shelf OCR systems. The TextVQA task aims to answer questions through reasoning over texts on images. Most conventional approaches fail to infer the right answer when the OCR system is imperfect. Instead, we propose a better OCR text encoding method and two pre-training tasks: masked language modeling (MLM) and masked bounding box prediction (MBBP) to help the model learn more robust OCR representations as well as strengthen positional reasoning. Through comprehensive experiments and analysis, we show that our model can overcome those limitations and achieve better performance on the position related questions compared with most of the baseline models.

1 Introduction

According to a VizWiz study (Bigham et al., 2010), 21% of questions asked by the visually impaired require reading and understanding text in the images. General Visual Question Answering (VQA) models perform poorly on these text-aware questions, as their model architectures do not contain any components that specialize in reading text in the image.

Singh et al. (2019) created the TextVQA dataset to address the text-aware VQA problems. They proposed the first TextVQA model LoRRA which uses pair-wise attention between question, image, and OCR (text detected in image) to predict a single-token answer. Hu et al. (2020) then introduced M4C where a multimodal transformer is used for the modality fusion in a common semantic space, and a dynamic pointer network is applied to predict multi-token answers. Yang et al. (2021) largely followed the framework of M4C and incorporated three unsupervised text-aware pre-training (TAP) tasks - masked language modeling, image-text matching and relative position prediction - to further lift the state-of-the-art performance on the TextVQA dataset. However, there are some common drawbacks of current TextVQA approaches. As shown in Figure 1, all of the systems drop significantly on the accuracy score when the OCR does not capture the ground truth answers, implying that current approaches are largely dependent on the OCR performance and very sensitive to the omissions caused by OCR system. Also, we observe that over 10% of the questions from TextVQA validation set involve reasoning about position relationships. Therefore, we are motivated to investigate ways to enhance the model's ability to deal with both inferior OCR system and position related questions.

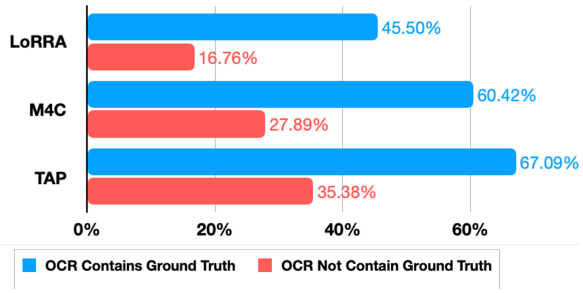


Figure 1: Model accuracy score over subsets in which OCR contains answer and OCR does not contain answer.

We introduce HotpotNet to overcome the limitation of the current OCR system in two ways. The first way is to supplement the model with other useful features as a remedy when OCR loses key information to answers. In this manner, we tried incorporating whole image features as well as object label text into the M4C framework. For pre-training, we performed the Masked Language Modeling (MLM)

*Everyone Contributed Equally – Alphabetical order

task in a way different from TAP aiming to recover the missing OCR and object labels. For fine-tuning, we tested a robust fine-tuning strategy so that the model could predict right answers without OCR. The second way is to enhance the current OCR system on both the input side and the encoding side. We replaced the current Rosetta OCR system with a better system, Microsoft Azure OCR. We also tried enhancing the text encoding of the OCR encoder by replacing the FastText embedding of OCR with BERT embedding. Furthermore, we proposed a new pre-training task called Masked Bounding Box Prediction to help enhance the model's performance on position related questions.

To sum up, our main contributions are:

1. We explored two ways to overcome OCR system limitations. First, we proposed a new MLM pre-training task and a robust fine-tuning method, as well as experimented with several new input features including whole image feature and object label text as a remedy for inadequate OCR. Second, we conducted experiments with a better OCR system and a better way of OCR text encoding.

2. For better performance on position-related questions, we proposed a novel Masked Bounding Box Prediction (MBBP) task, which helps strengthen the model's positional reasoning capabilities.

3. Experiment results show that our two pre-training tasks bring improvements compared with our baseline models. Also, we show the effectiveness of a better OCR system and our proposed way of OCR text encoding through comprehensive experiments and analysis.

2 Related Work and Background

2.1 Related Datasets

TextVQA, TextVQA-X, and TextCaps TextVQA[1] contains 28,408 images from OpenImages and 45,336 questions that require reading and reasoning about text in images. Each question comes with 10 ground truth answers. TextVQA-X (Rao et al., 2021) contains 11,681 images and 15,374 questions from TextVQA. For each question, up to 5 distinct human annotators provide visual and textual explanations for why a given answer is correct. TextCaps (Sidorov et al., 2020) is another dataset that is built upon TextVQA. 5 captions are collected for each image in the TextVQA dataset.

ST-VQA Created with a similar purpose as that of TextVQA, ST-VQA (Biten et al., 2019) comprises 23,038 images sourced from six datasets and 31,791 questions that can be unambiguously answered using text in the image. (In contrast, 39% of the answers in TextVQA do not contain OCR tokens.) Each question comes with up to 2 ground truth answers.

VQA v2.0 As an updated version of the first large-scale VQA dataset, VQA v2.0[2] contains 265,016 images and abstract scenes, 204,721 of which are COCO images. While there are a total of 1,105,904 questions, only 8k (or less than 1%) of these questions require reading text in the image (Biten et al., 2019). Each question comes with 10 ground truth answers.

OCR-CC Another large-scale dataset worth mentioning is the OCR-CC dataset (Yang et al., 2021). It contains 1.367 million scene text-related image-caption pairs from the Conceptual Captions dataset. The scene text detected per image has a mean and median of 11.4 and 6, compared with 23.1 and 12 in TextVQA, and 8.03 and 6 in ST-VQA.

VisualMRC Recently Tanaka et al. (2021) proposed the VisualMRC dataset which is a machine reading comprehension dataset based on document images. VisualMRC contains long abstractive answers which do not correspond to spans in the documents. Images in VisualMRC are sourced from multiple domains which makes it more challenging than DocVQA.

2.2 Relevant Techniques

2.2.1 Feature Extraction

Question Embedding Pretrained language models (Devlin et al., 2018) have become the dominant text encoders in recent work. These models are trained on massive amount of unlabeled text by minimizing unidirectional or bidirectional language modeling loss and later finetuned on specific domains.

Regional Image Feature Faster R-CNN model is the dominant approach used in VQA related tasks to extract region-based object features, including

[1] See https://textvqa.org/ for details.

[2] See https://visualqa.org/ for details.

visual information (convolutional features), positional information (bounding box coordinates) and class labels of detected objects. Apart from feature information on detected objects, there are papers such as Gao et al. (2020) that extract grid-based features as visual embedding, which mainly relies on ResNet and its variants to learn image representation as 2048-D vectors corresponding to 196 grids.

Whole Image Feature Rao et al. (2021) uses Feature Pyramid Network (FPN) to construct a semantic segmentation of the image and obtain visual explanations. In addition, although not mentioned in the VQA papers covered in this literature review, depth map is another potentially meaningful representation of the whole image besides RGB information, towards which Dense Prediction Transformer (DPT) (Ranftl et al. (2021)) is the state-of-the-art approach to estimate fine-grained depth information based on an architecture combining vision transformer encoder and convolutional decoder.

OCR System OCR methods used in prior work include Rosetta-en OCR, Rosetta-ml OCR, SBD-Trans OCR, and Google-OCR. While LoRRA (Singh et al., 2019) only uses FastText to extract word embedding from the OCR tokens, M4C and many later models employ a multi-feature representation, including appearance feature extracted using Faster R-CNN, character feature extracted using Pyramidal Histogram of Characters (PHOC), and a 4-dimensional location feature of the token's relative bounding box coordinates. In addition, Gao et al. (2021) introduced a 512D CNN feature called RecogCNN, which is extracted from text visual patches and trained on a text recognition task.

2.2.2 Multimodal Multitask Learning

There has been progresses made in the area of multimodal multi-task learning. Lu et al. (2020) jointly trained 12 vision-and-language tasks with a multi-task transformer based on VILBERT (Lu et al., 2019). Hu and Singh (2021) further expanded beyond fixed input modalities and jointly handled different single modal and multimodal tasks with a unified transformer model. They also pointed out the multimodal tasks such as VQA benefit from multi-task traning with uni-modal tasks.

Besides jointly training on tasks from different domains, several works designed a variety of highly correlated tasks to enhance the performance of VQA models.

Bounding Box Prediction Han et al. (2020) introduces a bounding box prediction task to prove its confidence of answer prediction. Specifically, when the model generates the answer by copying from OCR tokens, the IoU between the predicted bounding box and the ground truth OCR bounding box is calculated, serving as an evidence. The loss of the IoU is added into training loss to urge the model to predict credible answer.

ANLS as Reward Average Normalized Levenshtein Similarity (ANLS) is another popular metric in TextVQA task. It is used to measure the similarity between predicted answer and ground truth answer instead of binary comparison. It is common case that OCR system provides an incorrect token that model fail to get the exact correct answer due to the systematic error. ANLS guides the model to make the right prediction even when the predicted answer does not exactly match it ground truth. Many prior works(Zhu et al., 2020; Liu et al., 2020) introduced ANLS as a reward into the training loss.

2.2.3 Attention Mechanisms

Graph Attention Networks (GAT)(Veličković et al., 2017) is broadly used in prior works to encode visual relationship between objects, which has proven to be crucial to many computer vision tasks. ReGAT(Li et al., 2019a) introduces a Relation-aware GAT to model multi-type inter-object relations, including positional relation, semantic interactions and implicit relations. SA-M4C(Kant et al., 2020) improves M4C by introducing spatially aware self-attention layer where objects attend each other in a spatial graph.

Visual relation is important, though, TextVQA task requires a better understanding of relationship across multiple modalities. Many previous works explored relations between objects and OCR under the question text supervision. CRN(Liu et al., 2020) feeds question text features, OCR text features, and visual features into the Progressive Attention Module in turn and update informative features gradually. SSBaseline(Zhu et al., 2020) encodes multimodal features with three attention blocks, in each block OCR visual features, OCR text features, and object features attended with question embedding respectively. SMA(Gao et al., 2021) uses a Question Conditioned Graph Attention Module to

encoder the object-object, object-OCR, OCR-OCR relationships under the question's guidance.

2.2.4 Text-Aware Pre-training Tasks

Given that our target task requires reasoning over both text and images, our main focus here is the Vision-and-Language Pre-training (VLP). There are several strategies commonly used to train VLP models.

Image Text Matching This task requires the model to generate high-quality instance-level representations. Given a random pair of image and text descriptions, the model predicts whether the pair is matched. This task is widely used in a vartity of VLP models including ViLT, TAP, VISUALBERT, VILBERT, and LXMERT (Kim et al., 2021; Yang et al., 2021; Li et al., 2019b; Lu et al., 2019; Tan and Bansal, 2019). LayoutLM (Xu et al., 2020) further modified this task into Multi-label document classification task and used the document tags to supervise the document-level representation learning.

Masked Language Modeling Following the mechanisms proposed by Kenton and Toutanova (2019), this objective aims at predicting masked text tokens from the given contextualized vector and vectors corresponding to image regions. Kim et al. (2021) introduced whole word masking for MLM task to train VLP models, while Powalski et al. (2021) proposed to use T5-like salient span masking schema. Lu et al. (2019); Tan and Bansal (2019) extended this idea to masked multi-modal modeling task. In this manner, 15% of both words and image region inputs are masked and the model is required to reconstruct given the remaining inputs.

Other Cross-Modality Tasks Addition to the above prediction tasks, a few models also introduce several tasks that need strong cross-modality representations. Xu et al. (2021) proposed a text-image alignment task to encourage the model learn the alignment of detected objects among different modalities. Yang et al. (2021) designed a relative (spatial) position prediction (RPP) task. The RPP task aims to predict the relative spatial position between an object region and a scene text region. Tan and Bansal (2019) proposed image question answering tasks on the pretraining stage to further enhance the cross-modality representations.

2.2.5 Multireference

Rao et al. (2021) uses the sample one technique to leverage the multiple textual explanations collected for each question. In each training epoch, one of the available textual explanations is randomly selected.

2.2.6 Copy Mechanism and Pointer Network

Many sequence learning tasks requires *copying*, which refers to selectively replicating segments of the input to generate the output. For example, in a dialogue system, the agent needs to generate responds by referring to entities in the input utterance. Another example is text summarization, where the model is required to extract text from the original documents. Similar phenomenon is also observed in real-world language communication where humans tend to repeat long phrases in conversation.

Various pointer network (Vinyals et al., 2015) architectures have been proposed to address the challenge of *copying*. Gu et al. (2016) proposed CopyNet to address copying in seq2seq learning tasks. CopyNet generates an answer token at each timestep based on a mix of generation-mode probabilities and copy-mode probabilities, where the copy-mode probabilities are computed by attending to the input tokens. See et al. (2017) generalized CopyNet by modeling the selection between generation and copyping with a binary classifier.

Copying is also critical in text-related VQA tasks, where the answers often include OCR tokens in the image. LORRA (Singh et al., 2019) concatenates OCR tokens to the common word vocabulary and compute normalization over the new vocabulary. M4C (Hu et al., 2020) also augments transformer with a dynamic pointer network which computes OCR token probabilities with a bilinear layer.

3 Task Setup and Data

3.1 Task Setup

TextVQA is a VQA task that is specifically designed to evaluate models' abilities to reason about scene text in the image. Each sample in TextVQA consists a question and an image. The model is required to predict the correct answer based on the question and the image. The dataset also provides a set of objects tokens extracted from the image using faster-RCNN and a set of OCR tokens produced by an OCR recognizer.

The questions of TextVQA are diverse and answers are free-form natural language text with vari-

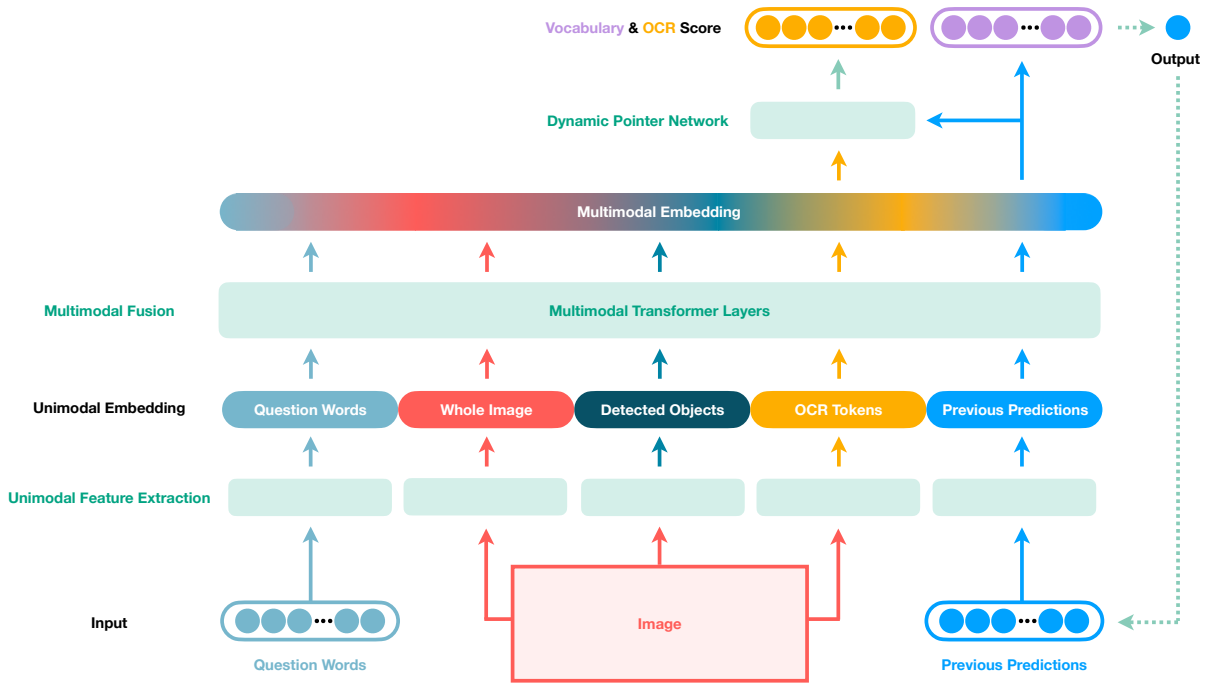


Figure 2: Model Architecture of HotpotNet

able number of word tokens, which makes it more difficult than multiple-choice question answering tasks where a small set of candidate answers are given.

3.2 Dataset Properties

TextVQA v0.5.1 contains 45,336 questions based on 28,408 images.

1. Training set contains 34,602 questions (103 MB) based on 21,953 images (6.6 GB) from OpenImages' training set.

2. Validation set contains 5,000 questions (16 MB) based on 3,166 images from OpenImages' training set.

3. Test set contains 5,734 questions (13 MB) based on 3,289 images (926 MB) from OpenImages' test set.

4 Baselines

4.1 Unimodal Baselines

LoRRA firstly explored unimodal models in TextVQA task. When only using the question module and predicting from the 8000 most frequent answers, LoRRA (Singh et al., 2019) achieves validation and test accuracies of 8.09% and 8.70%. When only using the image module and predicting from the 8000 most frequent answers, LoRRA achieves validation and test accuracies of 6.29% and 5.58%. LoRRA uses Rosetta-ml to produce OCR tokens. By predicting a random OCR token present in an image, validation and test accuracies of 7.72% and 9.12% are achieved. By predicting the most frequently occurring OCR token in an image, validation and test accuracies of 9.76% and 11.60% are achieved.

We include the following unimodal baselines: OCR, question text, and detected objects. All these unimodal baselines were run using M4C† architecture (as detailed in Section 4.2), and only OCR token embedding, or question word embedding, or detected object embedding was passed into the multimodal transformer layer.

Unimodal Baseline 1 (OCR) To embed N OCR tokens as $\{x_n^{ocr}\}$, we follow M4C's approach (Hu et al., 2020) and include FastText word embedding, Faster R-CNN appearance features, PHOC character features, and a 4-dimensional location feature:

$$x_n^{ocr} = LN(W_1 x_n^{ft} + W_2 x_n^{fr} + W_3 x_n^{p}) + \mathrm{LN}(W_4 x_n^{b})$$

where W_1, W_2, W_3, and W_4 are learned projection matrices and LN$(\cdot)$ is layer normalization. Differ-

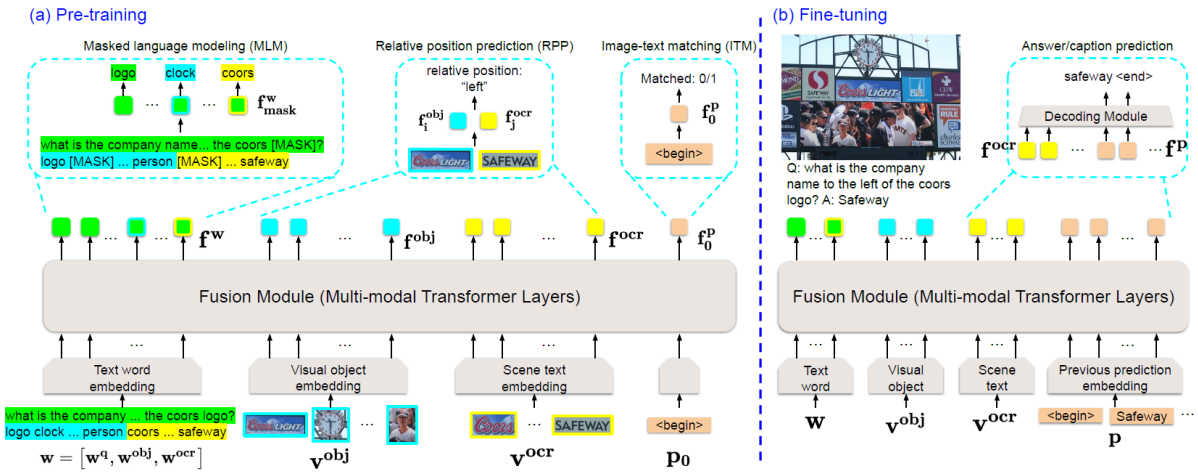


Figure 3: Model Architecture of TAP

ent from M4C, the OCR system used is Microsoft Azure OCR. We expect the OCR baseline to capture information provided by scene text that is relevant for answering a given question.

Unimodal Baseline 2 (Question text) To embed K question words as $\{x_k^{ques}\}$, we follow M4C's approach and use a pretrained BERT model. We expect the question text baseline to capture information from questions.

Unimodal Baseline 3 (Detected objects) We obtain M detected objects through Faster R-CNN, as in M4C. In addition to an appearance feature and a 4-dimensional location feature used by M4C, we incorporate label text, using FastText, into $\{x_m^{obj}\}$:

$$x_m^{obj} = LN(W_5 x_m^{fr} + W_6 x_m^{ft}) \; + \; LN(W_7 x_m^{b})$$

where W_5, W_6, and W_7 are learned projection matrices and LN$(\cdot)$ is layer normalization. We expect the detected objects baseline to capture information from image objects that helps answer a given question.

4.2 Simple Multimodal Baselines

M4C† The architecture is the same as M4C (Hu et al., 2020) but **M4C†** changed the OCR system from Rosetta-en to Microsoft Azure OCR for better performance.

M4C†w/o MMT In **M4C†** architecture, question text, detected object, OCR token, and previous prediction tokens are encoded with a deep multimodal transformer. With encoded multimodal features, a classifier is used to calculate scores for tokens in fixed vocabulary and a dynamic pointer network is used to dynamically calculate scores for OCR tokens from the image. In this baseline, we explored a simple attention network to model both inter- and intra-modality relations as an alternative for the multimodal fusion layers. Following previous works, we define the *decoding output* as the previous predictions. The encoded decoding output, h^{dec} serves as the input of classifier. Both encoded decoding output, h^{dec}, and encoded OCR representation, h^{ocr}, together serve as the input of DPN. In HotpotNet, a deep unified multimodal transformer encoder is used to get both of the encoded features simultaneously. In this simple baseline, we encode them separately in a pair-wise way.

Three multi-head self attention modules are firstly used to separately model the interaction between previous prediction x^{dec} and each modality embeddings (x^{ocr} for OCR; x^{obj} for detected objects; x^{ques} for question text):

$$z^{dec\_ques} = \text{SelfAttention}([x^{dec}; x^{ques}])$$
$$z^{dec\_ocr} = \text{SelfAttention}([x^{dec}; x^{ocr}])$$
$$z^{dec\_obj} = \text{SelfAttention}([x^{dec}; x^{obj}])$$

Then the outputs of each self attention module are summed up to get the simple multimodally encoded decoding representation:

$$h^{dec} = z^{dec\_ques} + z^{dec\_ocr} + z^{dec\_obj}$$

The multimodal encoding of OCR token representations is similar:

$$z^{ocr\_ques} = \text{SelfAttention}([x^{ocr}; x^{ques}])$$
$$z^{ocr\_obj} = \text{SelfAttention}([x^{ocr}; x^{obj}])$$
$$h^{ocr} = z^{ocr\_ques} + z^{ocr\_obj}$$

During decoding, for each time step t, the fixed score for the vocabulary is calculated by a classifier:

$$y_{t,m}^{\text{vocab}} = \left(w_m^{\text{vocab}}\right)^T h_t^{\text{dec}} + b_m^{\text{vocab}}$$

where w_m^{vocab} is a d-dimensional parameter for the m-th wordx.

The copy score for the dynamic pointer network is calculated through the decoding outputs h_t^{dec} and OCR token representations h_n^{ocr}:

$$y_{t,n}^{\text{ocr}} = (W^{\text{ocr}} h_n^{\text{ocr}} + b^{\text{ocr}})^T \left(W^{\text{dec}} h_t^{\text{dec}} + b^{\text{dec}}\right)$$

where W^{ocr} and W^{dec} are $d \times d$ matrices, and b^{ocr} and b^{dec} are d-dimensional vectors.

M4C†w/o DPN For the third simple multimodal baseline, we replaced the dynamic pointer network (DPN) decoder of HotpotNet with the classifier used in LoRRA (Singh et al., 2019). In our implementation, we passed the decoding outputs into the LoRRA's classifier to either select a frequent answer from the training set or copy a single OCR token in the image as the answer.

4.3 Competitive Baselines

We reproduced the performance of three competitive baselines that were previously proposed in research papers, namely LoRRA, M4C and TAP. In this section, we will briefly introduce their model architecture and key insights.

LoRRA Singh et al. (2019) proposed LoRRA that is specifically designed for reasoning about text in images. They proposed a reading module that uses an OCR model to extract word tokens from the image, embeds them with FastText vectors and computes contextual attention based on the question to get the combined representation.

$$f_{OCR}(s, q) = f_{comb}(f_A(f_O(s), f_Q(q)), f_Q(q))$$

LoRRA predicts answer by selecting from a vocabulary of frequent answers and the OCR tokens in the image. Experiments show that the proposed reading module, when used with the previous state-of-the-art VQA model, improves performance significantly on TextVQA and slightly on VQA 2.0.

M4C The major limitation of LoRRA is that it only models interaction between pairs of modalities with pairwise fusion mechanism. M4C (Hu et al., 2020) addressed this with a multimodal transformer architecture that enables fusion of more than two modalities in a common semantic space. They also replaced the answer prediction modules with a dynamic pointer network to enable prediction of multi-token answers.

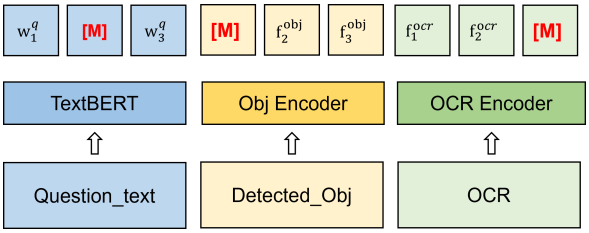


Figure 4: An illustration of our MLM for HotpotNet. [M] denotes the masked inputs. For masked OCR and object inputs, we mask both the text and visual features and keep the location features.

TAP Yang et al. (2021) further explored unsupervised text-aware pre-training (TAP) with masked language modeling, image-text matching and relative position prediction, and experimental results show that pre-training on TextVQA itself is able to boost performance by $+5.4\%$. The architecture of TAP (Fig 3) is similar to M4C except that the former further includes object labels.

5 Proposed Model

In this section, we will briefly introduce our proposed model **HotpotNet** and our pre-training and fine-tuning strategies. The overall architecture is shown in Figure 2. Our framework is the same as M4C† except that we incorporate two additional features, object label text and whole image features. Based on our experiment results, we do not use the whole image features in our final implementation. The detailed analysis of the whole image feature is in Section 7.

5.1 Model Architecture

For our first proposed method, **HotpotNet (FastText)**, no pre-training task is performed, and we use FastText to embed both OCR tokens and detected object labels. The only difference between HotpotNet (FastText) and M4C† is that we further incorporate the label text features into the detected object features. We expect this new feature to improve performance over the questions that mention objects in the image.

For our second proposed method, **HotpotNet (BERT)**, no pre-training task is performed, and we

replace the FastText word embedding of OCR tokens and detected object labels in HotpotNet (FastText) with BERT word embedding. Since the question words are encoded by BERT, we are motivated to replace the FastText embedding of both object labels and OCR tokens with BERT embedding to enhance the correlations between question words and OCR/ detected objects.

For our final proposed method, **HotpotNet (BERT) + Pre-Training**, we further incorporate the pre-training strategies as detailed in Section 5.2 into HotpotNet (BERT).

5.2 Pre-training

We propose to perform the Masked Language Modeling (MLM) task and a novel Masked Bounding Box Prediction (MBBP) task to pre-train the model.

Masked Language Modeling (MLM) Tokens from question text, detected object label, and OCR token is randomly masked with a probability of 15%. Following Kenton and Toutanova (2019), we then replace the masked word with a special `<MASK>` token 80% of the time, replace it with a random word 10% of the time, and do not change the word 10% of the time. This task aims to recover the masked word token. Notably, our MLM task is largely different from TAP's. In TAP, the question text input is extended with OCR tokens and object label text, and MLM is performed on the extended text input $\mathbf{w} = [\mathbf{w^q}, \mathbf{w^{obj}}, \mathbf{w^{ocr}}]$. The system needs to learn the alignment between text inputs and OCR/object features to recover the masked input. We argue that by directly combining the text feature with OCR/object features, the system no longer needs to learn the alignment and therefore MLM can be performed more efficiently. As shown in Figure 4, instead of building up extended text inputs with object label text and OCR text, we propose to directly mask both visual and text features and only keep the location features in the object feature and the OCR feature and task the system with recovering the corresponding text from the masked features.

Masked Bounding Box Prediction (MBBP) We randomly mask the bounding box coordinates of detected objects or OCR tokens with a probability of 15%. The masked coordinates are replaced with 0. This task aims to recover the original position coordinates:

$$\mathbf{pos} = [\frac{x_{\text{lu}}}{W}, \frac{y_{\text{lu}}}{H}, \frac{x_{\text{rb}}}{W}, \frac{y_{\text{rb}}}{H}]$$

where W, H denotes the width and height of the image, $(x_{\text{lu}}, y_{\text{lu}})$ and $(x_{\text{rb}}, y_{\text{rb}})$ represent the coordinates of the left-upper point and the right-bottom point of the bounding box respectively.

5.3 Loss Functions

During pre-training, we utilize the same TextVQA training set as our pre-training corpus. As described in Section 5.2, we proposed a loss as the sum of two auxiliary losses for MLM and MBBP tasks respectively during unsupervised pre-training:

$$\mathcal{J}_{\text{pretrain}} = \sum_{\mathbf{x}} \lambda_{\text{MLM}} \mathcal{J}_{\text{MLM}}(\mathbf{x}) + \lambda_{\text{MBBP}} \mathcal{J}_{\text{MBBP}}(\mathbf{x})$$

where $\mathcal{J}_{\text{MLM}}$ is cross-entropy loss, and $\mathcal{J}_{\text{MBBP}}$ is L1 loss. λ_{MLM} and λ_{MBBP} are weights.

After pretraining, we directly finetune our model on the TextVQA dataset, we train our model by minimizing the cross-entropy loss. The loss is calculated between the predicted answer and the ground-truth answer. Our loss function can be formally written as follows:

$$\mathcal{J}_{\text{finetune}} = \sum_{\mathbf{x},\mathbf{y}} \mathcal{L}(\mathbf{x}, \mathbf{y})$$

where $\mathcal{L}$ denotes the cross-entropy loss.

5.4 Hyperparameters and Their Effects

The hyperparameters are described in Table 1. In pre-training, we use weights 1 and 0.001 for MLM and MBBP respectively. This weight ratio ensures that the two tasks start with losses of the same magnitude and thus encourages the model to pay comparable attention to the two tasks.

6 Results

6.1 Evaluation Metric

Following previous work (Hu et al., 2020; Yang et al., 2021), we evaluate the models using an accuracy metric based on soft voting. TextVQA provides 10 human-annotated answers for each questions. The accuracy score of a certain answer in a certain subset is computed by averaging the following metric over all 9 out of 10 subsets of annotators.

$$\text{Acc}(ans)_i = \min\left\{\frac{\text{\#humans that said } ans}{3}, 1\right\}$$

Hyper-parameter	Value
(a) General parameters	
max length of question word w^q	20
max length of visual object f^{obj}	100
max length of scene text f^{ocr}	100
optimizer	Adam
batch size	128
base learning rate	1e-4
warm-up learning rate factor	0.2
warm-up iterations	2000
max gradient L2-norm for clipping	0.25
learning rate decay	0.1
(b) Pre-training parameters	
learning rate steps	14K, 19K
max iterations	24K
mask rate for OCR, object, question text	0.15
MLM loss weight λ_{MLM}	1
MBBP loss weight λ_{MBBP}	0.001

Table 1: Hyper-parameters of the HotpotNet experiments.

The final accuracy score of a certain answer is the average of 10 subsets scores.

$$\text{Acc}(ans) = \frac{1}{10}\sum_{i=1}^{10}\text{Acc}(ans)_i$$

The evaluator also performs text normalizations such as lower-casing and converting number words to digits before evaluating the answers [3].

6.2 Results Comparison

6.2.1 Proposed Method Comparison

Effect of BERT encoding Compared with HotpotNet (FastText), HotpotNet (BERT) performs 1.2% better on accuracy score. This shows that enhancing the text encoding for object label and OCR tokens further boosts the model's performance.

Effect of pre-training HotpotNet (BERT) + Pre-training further outperforms HotpotNet (BERT) by $+0.5\%$ accuracy. This shows that even pre-training on the same dataset used in fine-tuning can improve the model's performance.

6.2.2 Proposed Methods versus Previous Approaches

Q1: How does detected object label text effect the system's performance?

Compared with **M4C**†, HotpotNet(FastText) incorporates the detected object label text but the performance drops slightly. This implies that the model is aware of the detected object labels after directly fine-tuning and no longer needs explicitly adding the object labels.

[3]For full details of the evaluation, please refer to https://visualqa.org/evaluation

Methods	**Accuracy (%)**
Unimodal Baselines	
OCR only	25.26
Question text only	19.00
Detected objects only	12.72
Simple Multimodal Baselines	
M4C†	45.41
M4C† w/o MMT	41.38
M4C† w/o DPN	28.88
Previous Approaches	
LoRRA (Singh et al., 2019)	27.57
M4C (Hu et al., 2020)	39.23
TAP (Yang et al., 2021)	49.26
TAP w/o Pre-Training	44.50
Proposed Methods	
HotpotNet (FastText)	45.32
HotpotNet (BERT)	46.54
HotpotNet (BERT) + Pre-training	47.04

[1] M4C†: M4C with Rosetta OCR replaced by Microsoft OCR

Table 2: Experiment results of unimodal, multimodal, competitive baselines, and proposed methods on TextVQA dev set (Singh et al., 2019).

Q2: Is our pre-training task better than TAP's?

By comparing the results of HotpotNet (BERT) + Pre-training with TAP, we can conclude our way of pre-training is not better than TAP's. Our pre-training task of MLM directly uses the alignment information between text and visual features for object and OCR and perform masking over the whole sequence including question text, OCR, detected objects. TAP extends question text inputs with OCR tokens and detected object labels and performs masking over the extended question texts. In other words, our pretraining task directly utilizes the alignment between text and visual features while TAP's method requires the model to infer the alignment between the text and visual features. This key difference and results imply that learning the alignment between text and visual features is essential in TextVQA performance.

Input Modality			Robust Fine-tuning	Accuracy(%)		
Question Text	OCR	Whole Image		Overall	$a \in OCR$	$a \notin OCR$
✗	✗	ViT	✗	5.84	3.59	8.51
✓	✗	✗	✗	12.85	9.18	17.35
✓	✗	ViT	✗	14.08	10.68	17.96
✓	Microsoft-OCR	✗	✗	44.06	60.38	30.01
✓	Microsoft-OCR	ViT	✗	43.70	59.87	29.57
✓	Microsoft-OCR	ViT	modality masking ($p = 0.05$)	40.72	57.54	26.82
✓	Microsoft-OCR	ViT	modality masking ($p = 0.2$)	40.42	58.18	24.28

Table 3: Ablation Study on Whole Image Features and Robust Fine-tuning

Mask Ratio of Pre-training tasks		Weights of Pre-training tasks		Accuracy(%)
MLM	MBBP	MLM	MBBP	
Fine-tuning Only				46.54
0.15	✗	1	✗	46.67
✗	0.15	✗	1	46.87
0.15	0.15	1	0.001	**47.04**

Table 4: Ablation Study on Pre-training.

7 Analysis

7.1 Ablation Study

7.1.1 Whole Image Features and Robust Fine-tuning

We perform experiments to study the impact of adding whole image visual features extracted from a pretrained image encoder.

We use `ViT_b_16` pre-trained on ImageNet, which is a transformer-based model introduced in Dosovitskiy et al. (2020). We resized all images to 224 by 224, and extracted visual features after the 5th transformer layer (there are 12 in total). Our choice of layer number is based on the intuition that low-level feature extracted after bottom layers (e.g. 1 and 2) may not reveal the connection between various image patches, and high-level feature extracted after top layers (e.g. 11 and 12) may be well adapted to image classification tasks and not detection tasks as desired in the TextVQA setting. As for our choice of pre-trained model, due to the nature of convolution, feature extracted from intermediate layers of models such as VGG (Simonyan and Zisserman (2014)) and ResNet (He et al. (2016)) may well fail to represent the inner relationship between image regions.

We further experimented with applying robust fine-tuning by modality-specific masking. For each training sample, we randomly choose to mask either the object modality or the OCR modality and randomly mask $x\%$ of the objects (or OCR tokens) by setting their features to zeros. The model is required to predict the ground truth from the masked features. We fine-tune the model by minimizing cross-entropy loss as in previous work (Hu et al., 2020; Yang et al., 2021). We believe this masking strategy will encourage the model to utilize information in all modalities and thus reduce the risk of overfitting a particular modality.

All the results are summarized in Table 3. For each model variant, we report its accuracy on the entire validation set as well as accuracy on a subset of validation samples where the ground truth is contained (or not contained) in the OCR tokens (denoted by $a \in OCR$ and $a \notin OCR$).

Results show that adding whole image features bring an improvement of $+1.23\%$ when OCR tokens are not present and an improvement of $+1.50\%$ on the $a \in OCR$ subset. However, adding these visual features does not bring improvement when the OCR tokens are available, indicating that whole image features fail to provide complementary information to the OCR tokens. We suspect that this is related to the pre-training task that ViT is trained on. Typical pre-training tasks like image classification do not requires scene text information to be encoded in the features. As a reulst, ViT fails to produce good representation about the scene text and their positions in the image. Similar phenomenon has also been observed in previous work (Biten et al., 2021).

Results also show that incorporating robust fine-tuning degrades performance by $-2.98\%(p =$

0.05). We suspect that masking token features to 0 introduces distribution discrepancy between training and inference.

7.1.2 Pre-training Tasks

We also implemented ablation study on pre-training tasks, *i.e.,* Masked Language Modeling (MLM) and Masked Bounding Box Prediction (MBBP), with model performance summarized in Table 4. We confirmed the effectiveness of both MLM and MBBP and found that performing both tasks achieved the best performance.

Compared with the fine-tuning only baseline of 46.54, MLM and MBBP, when performed alone, both lead to a slight improvement in accuracy (46.67 and 46.87). We then combined the two tasks with weights of 1000:1 to position the two losses at the same scale. Performing both tasks led to the most significant improvement in all pre-training ablations, from 46.54 to 47.04.

7.2 Intrinsic Metrics

In Figure 5(a), we visualize the overall distribution of accuracy scores on 5000 questions for each model, from which selected qualitative examples will be presented in Section 7.3. As presented in the bar plot, while the state-of-the-art TAP model tops all baselines in all categories, i.e. has the smallest number of 0-scored questions and the largest number of none-zero scored questions, our proposed methods (and the variant without object label input) also achieved leading performance, despite slightly fewer counts in predictions with 1.0 accuracy. Meanwhile, unimodal baselines as well as multimodal baselines such as LoRRA and M4C† w/o DPN have unarguably weaker performance. We extracted subsets of interest from validation data to further dissect model performance in the following subsections.

7.2.1 Performance on Position-related Questions

As mentioned in SA-M4C (Kant et al., 2020), $\sim 13\%$ of questions in the TextVQA dataset are related to spatial positions. Therefore, the capability to reason about positional information is essential in tackling TextVQA tasks. Accordingly, we sifted **position-related subset** of 540 questions from validation dataset where text prompt of each question explicitly contains at least one word from the predefined positional vocabulary. It is worth noting that the positional vocabulary consists of not merely simple nouns or prepositions such as `<left, right, under, upper, below, above, head, tail>`, but also more complex expressions denoting positional semantics such as "pointing to" and "the back".

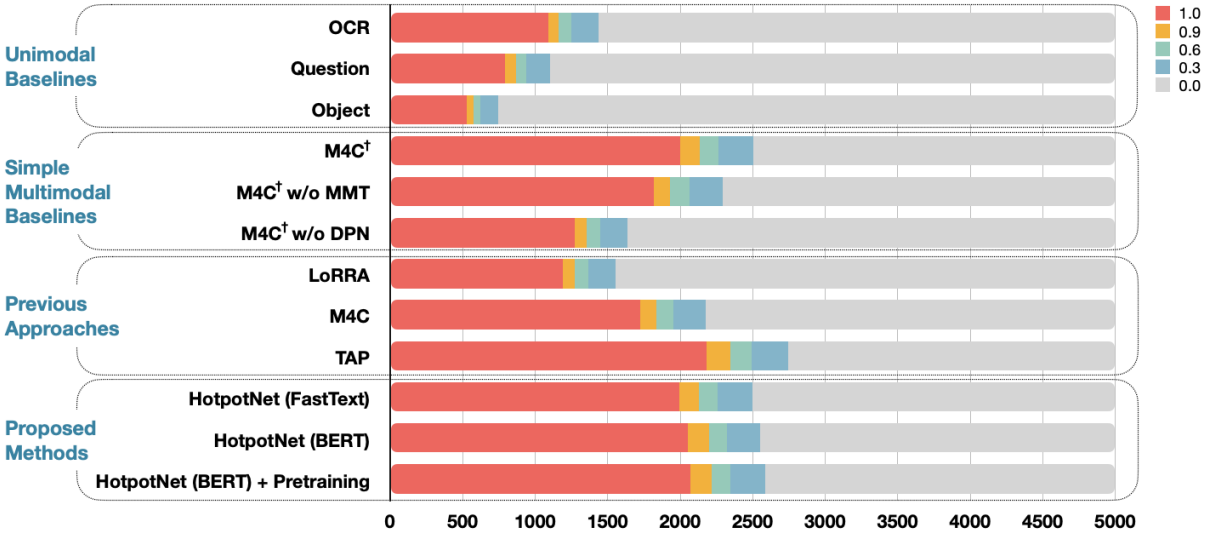
We visualized the distribution of accuracy scores in Figure 5(b). In **position-related subset**, although all models examined perform worse on the position related questions, the ratio of non-zero scores achieved by our proposed methods didn't shrink that much as baseline methods or LoRRA did. Moreover, the gap of accuracy distribution between HotpotNet (BERT) with pre-training and TAP becomes much smaller, denoting that the introduction of MBBP pre-training task may well have contributed to HotpotNet (BERT) achieving comparable performance as TAP did in positional reasoning.

7.2.2 Robustness against OCR System

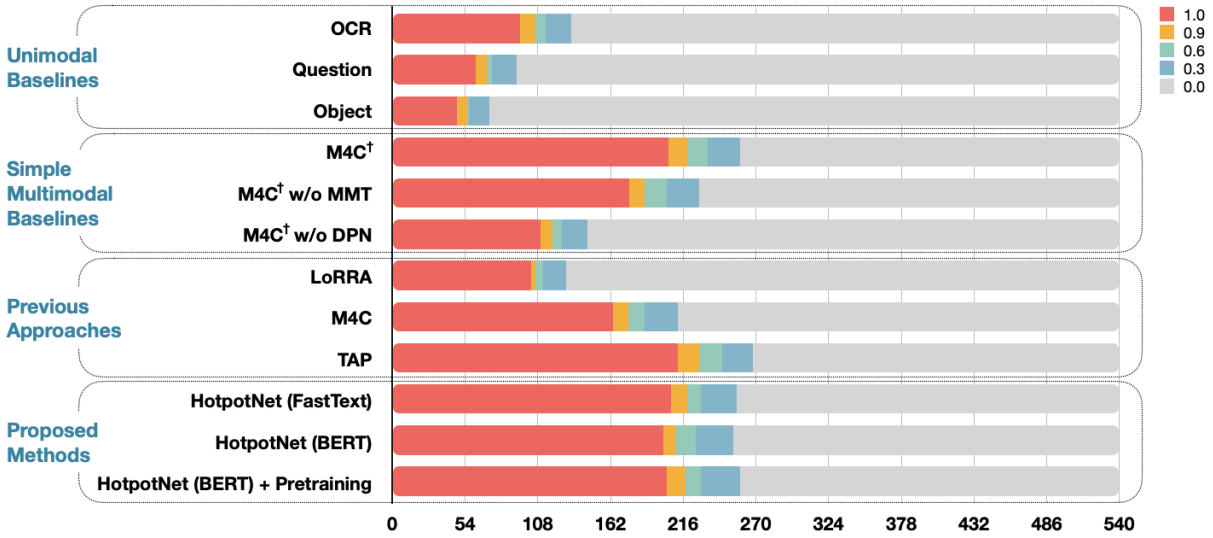
Since a fundamental portion of TextVQA tasks lies in reading and understanding textual information present in question images, the quality of OCR detection is essential to model performance. In this sense, how TextVQA models make predictions may well be prone to the failure of pre-trained OCR module. In fact, as demonstrated in Figure 1 and Table 3, for the same model structure, the prediction accuracy differs by up to 33.9% between the settings where ground truth token(s) are or are not present in OCR detection result.

Therefore, from our perspective, assuring model robustness against the performance of a pre-trained OCR module is key in solving TextVQA problems. Ideally, a model is expected to predict the correct answer despite the lacking of relevant OCR input. In our experiment, the evaluation of model robustness against OCR detection result is conducted by comparing model performance on two subsets of the validation dataset, one of which contains questions where neither Microsoft-OCR nor Rosetta-en manages to capture any ground truth answer token, while the other includes questions where both OCR modules successfully detect at least one ground truth answer token.

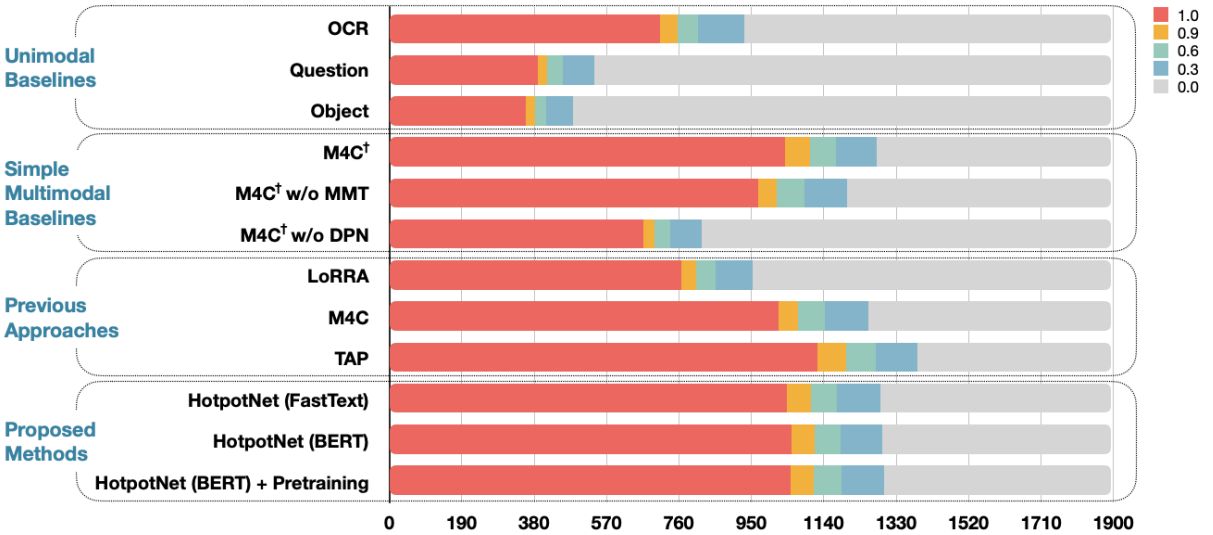
As presented in Figure 5(c) and Figure 5(d), the number of nonzero scored questions dramatically diminished for all TextVQA models of all groups, indicating an universally strong dependency of currently experimented models on OCR module. In particular, we are interested in comparing top accurate models in terms of robustness against OCR



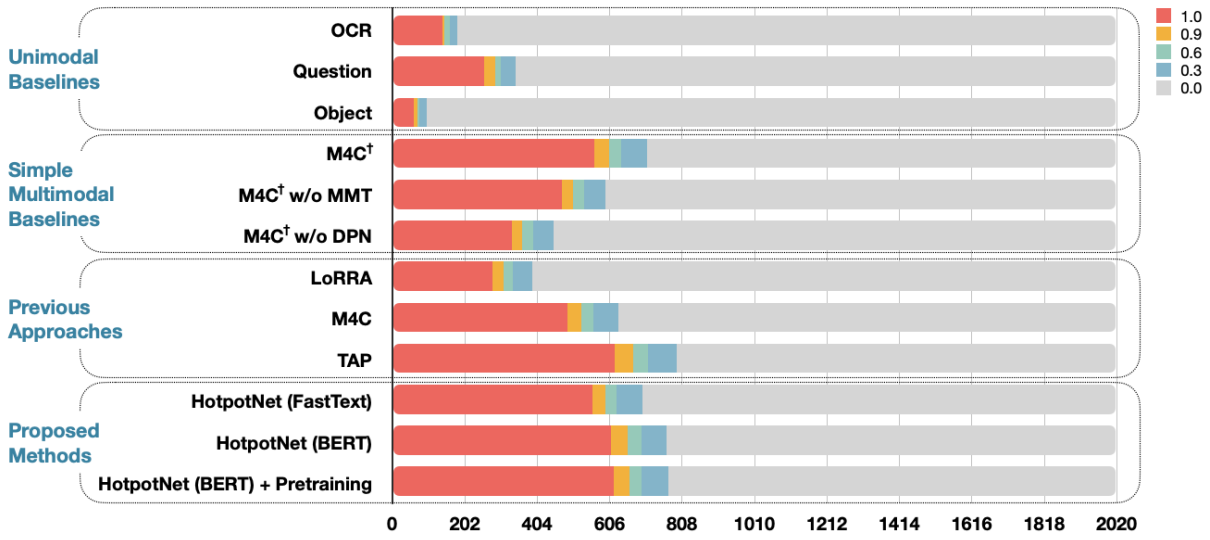
(a) Whole validation set



(b) Questions related to position



(c) Ground truth in OCR detection



(d) Ground truth not in OCR detection

Figure 5: Accuracy Distribution

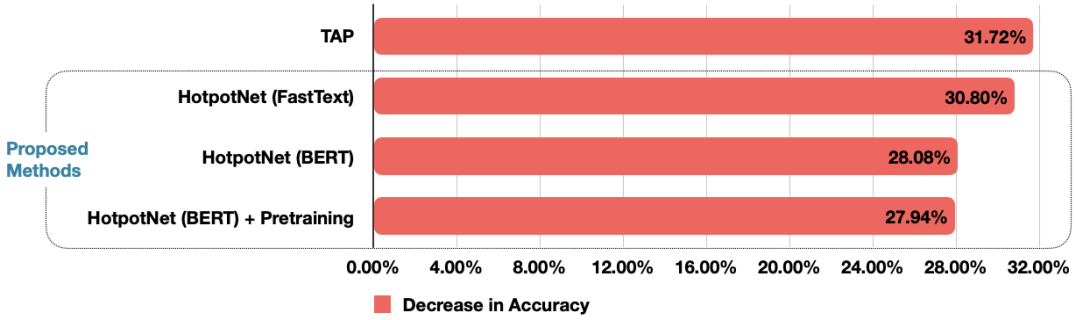


Figure 6: Accuracy Decrease Caused by OCR Not Containing Ground Truth

detection, and thereby visualize the accuracy declination of our proposed methods in juxtaposition with TAP. Figure 6 reveals a smaller shrinkage in accuracy of HotpotNet(BERT) + Pretraining (27.95%) compared with TAP (31.72%), illustrating relatively less reliance of HotpotNet(BERT) + Pretraining on OCR module.

For each model, we count the number of questions for which it got a 0 accuracy score while the others got nonzero scores.

7.3 Qualitative Analysis and Examples

In this section, we focus on the comparison of our proposed methods (HotpotNet (FastText), HotpotNet (BERT), and HotpotNet (BERT) + Pre-training (mentioned as "our final model" in the following analysis)) and TAP, which is the SOTA TextVQA model. We first qualitatively examine model performance by analyzing representative cases where our proposed model structure and pre-training approaches effectively assisted in addressing the questions, as well as where our model performed inferiorly. Then we summarize findings on subsets of questions that are deemed as typically challenging.

7.3.1 How does HotpotNet Help?

Position-related Questions Our novel pre-training task MBBP helps address position-related questions, a set of questions that are important but difficult for our baseline models to answer. As in the top question in Figure 7, our model predicted the correct answer "krainerwurst" copied from OCR token, whereas TAP and two HotpotNet variants without pre-training failed. In this particular position-related case, the text prompt explicitly denoted "top" and "left", which is crucial in answer reasoning and prediction. Although TAP performs the *relative (spatial) position prediction* (RPP) task to acknowledge model of positional information, RPP task primarily focuses on relative positions between objects and OCR tokens rather than positions

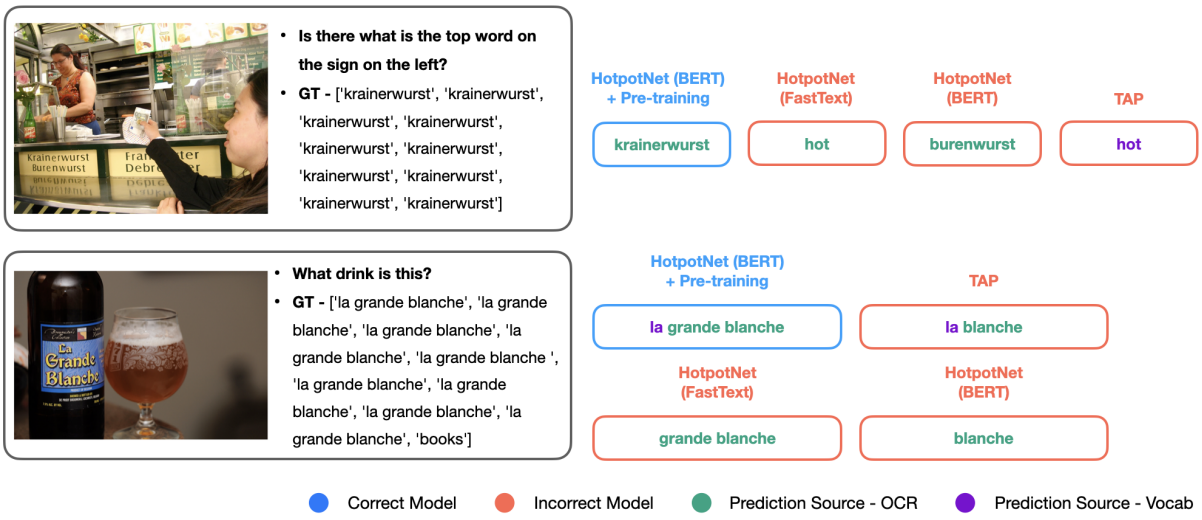


Figure 7: Previous Approaches Failure Cases

of OCR/object detection boxes in the whole image. By contrast, our proposed MBBP pre-training task reinforces the capability of HotpotNet in locating an OCR/object in the entire image and leveraging the positional information in addressing problems, leading to the success in this question.

Robustness against OCR System In TextVQA tasks, as demonstrated in quantitative results in Section 7.2.2, models would typically fail to predict a particular answer token that shows up in the image but is not captured by the pre-trained OCR module. Accordingly, we introduce the MLM pre-training task in an effort to promote multimodal interaction to infer the expected OCR token that is missing in the input. In the bottom question of Figure 7, despite having predicted the partially correct tokens "grande" and "blanche", HotpotNet (FastText) and HotpotNet (BERT) failed to add "la", which is desired for a contextually meaningful answer but not detected as OCR token. In comparison, through the synergy of existing OCR tokens as well as feature information from multiple other modalities, models pre-trained with MLM tasks are capable of predicting "la" from VOCAB, suggesting the positive impact of MLM pre-training task on model robustness against OCR detection failure.

7.3.2 HotpotNet Failure Cases

Positional Relation between OCR Tokens and Objects Apart from questions asking for positional relation between OCR/object and the whole question image, there are cases that require the model to figure out the relative position between objects and OCR tokens. The top question of Figure 8 shows an example where TAP outperforms HotpotNet variants thanks to its ability to recognize which OCR tokens are *on* the same detected object. In contrast, our proposed methods are confused by the many OCR tokens that are close to each other and select OCR tokens from different objects (the book on the bottom and the book second from the bottom). We hypothesize that the RPP task strengthens TAP's capacity to identify which OCR tokens should be pieced together by reasoning about the relative positions between OCR tokens and objects, and therefore manages to predict an answer consisting of the correctly aligned OCR tokens ("the", "speaking", and "eye" on the second from the bottom book).

The positional cases presented here and in Section 7.3.1, to some extent, delineate complementary relation between our proposed MBBP pre-training task and the RPP task, as the two tasks contribute to different aspects of the models' positional reasoning ability.

In-sequence Contextualization The bottom question in Figure 7.3.1 shows another HotpotNet failure case where TAP correctly answers the name of the book ("secrets of a ruthless tycoon") but our final model answers the author's name ("cathay williams"). Our interpretation is that the architectural difference between our final model and TAP in the input text sequence may have led to the outcome. TAP concatenates question texts with object labels and OCR tokens into one sequence as the input of the BERT textual encoder, while our final model uses BERT to encode question texts,

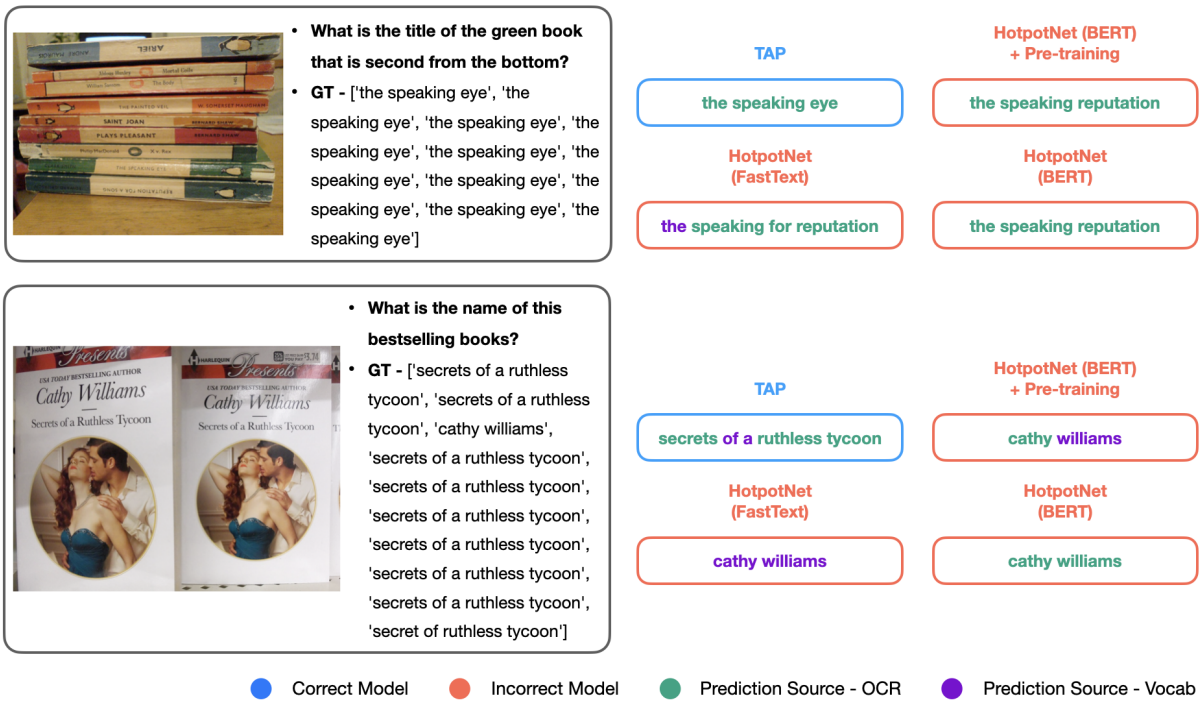


Figure 8: HotpotNet Failure Cases

object labels, and OCR tokens separately. As a pre-trained model excelling at contextualized language understanding, BERT is able to comprehend the relationship between question texts, object class label texts, and OCR tokens if they are in the same input sequence. In comparison, due to the isolation of texts from different modalities (question, object and OCR), our final model fails to take advantage of BERT's contextualized embedding of texts, and consequently fails to discern that "cathay williams" does not fit in the description of "the name of the bestselling books".

7.3.3 Challenging Cases

In addition to questions where either TAP or our final model performs poorly, we are interested in the case where all experimented models fail, i.e, we are interested in the "hard" questions. In the following paragraphs, we categorize challenging questions where none of the models outputs predictions matches any ground truth answer into two groups, and provide interpretation with reference to qualitative examples.

Multi-stage Reasoning The first group of challenging cases are those requiring multi-stage reasoning. As in Figure 9, model is asked whether all 8 clocks are set for the same time. To derive the correct answer, models are supposed to be able to firstly figure out the time of each clock and then compare the results over all the clocks. Similarly, the question of brewery requires models to not merely detect the brewery of each beer, but also compare the breweries among all beer to solve the puzzle.

Prior Knowledge Furthermore, there are questions where specialized background knowledge is essential to the answers but cannot be learned from question text or image. Although a model has the potential to learn common background knowledge during pre-training, more domain-specific knowledge implied by questions would still render the model impotent. As illustrated in the two examples in Figure 10, neither "roman numerals" nor "lottery" is present in the images, and presumably these concepts are rarely appear in the TextVQA dataset. Therefore, none of our experimented methods would be capable to predict correct answers as guided by question text "roman numerals" and "lottery".

8 Limitations and Future work

8.1 Limitations

8.1.1 Weak Performance on Positional Reasoning

Despite performance improvement in inferring and reasoning about positions of OCR/Object, as

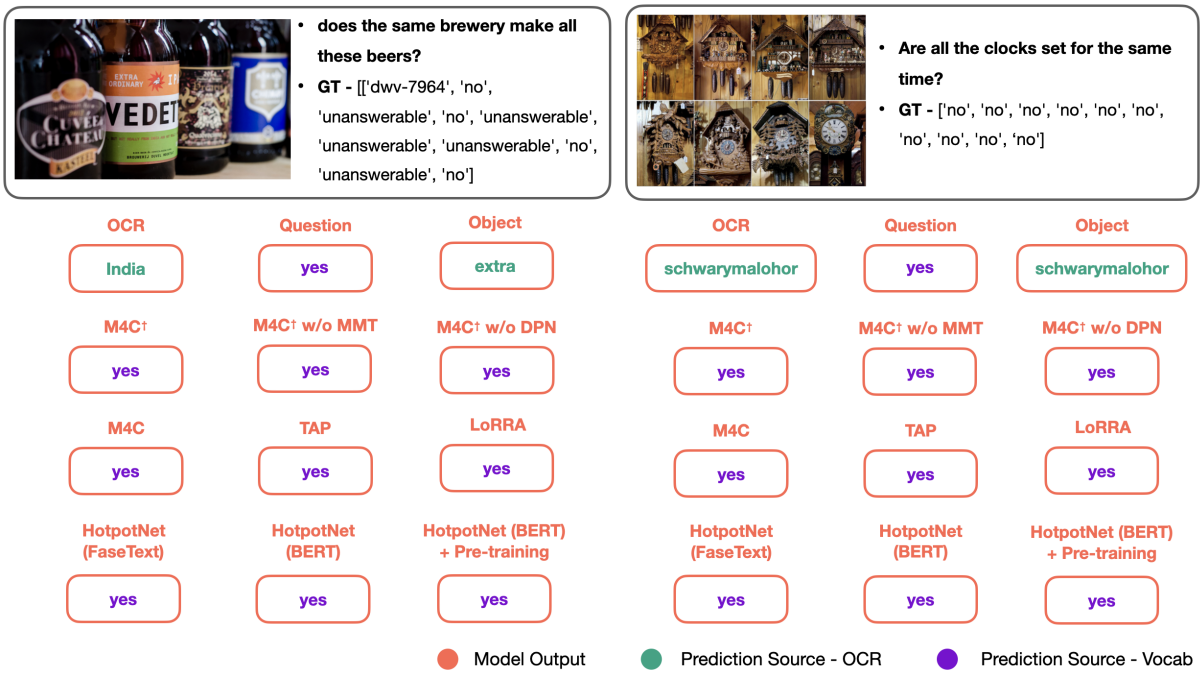


Figure 9: Challenging Case: Multi-Step Reasoning

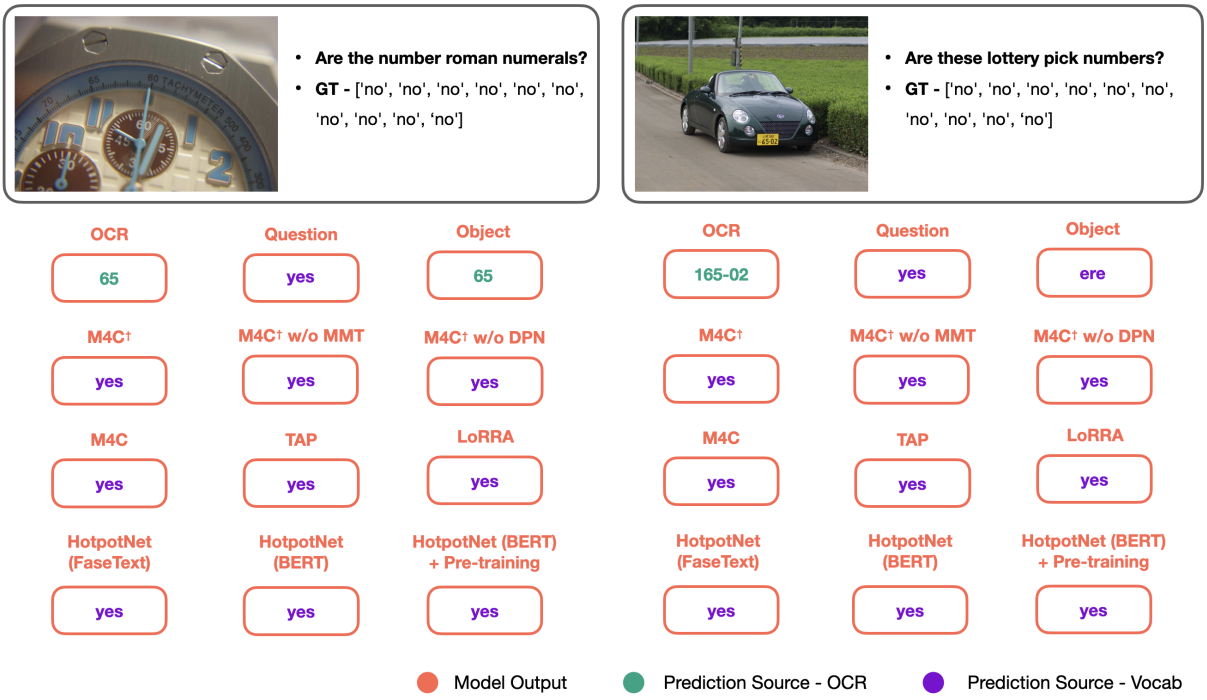


Figure 10: Challenging Case: Prior Knowledge

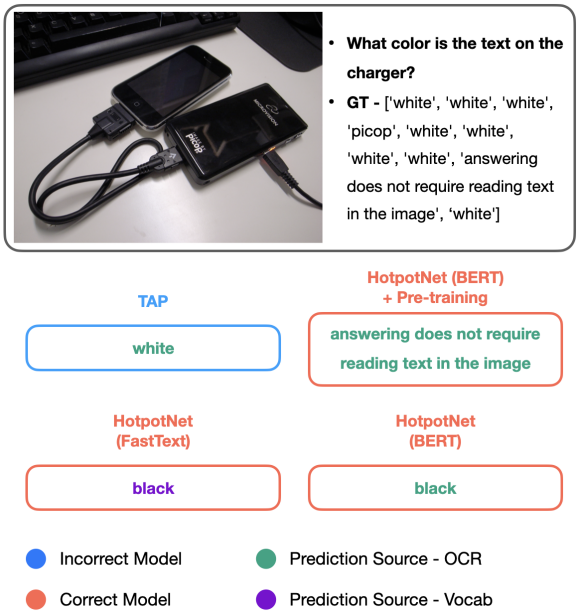


Figure 11: Example: Questions Not Requiring Reading Text

demonstrated in Section 7.3.2, our proposed methods underperform in questions that require the recognition of relative positions between OCR tokens and objects.

8.1.2 Strong Reliance on OCR System

For all proposed TextVQA model, an off-the-shelf OCR system is still a necessity. Better OCR system can always largely improve the model performance without anything else changed. As expounded in Section 7.2.2 and Section 7.3.1, although our proposed MLM pre-training task proves to improve the model robustness to OCR system, our final model (HotpotNet (BERT) + Pre-training) still suffers from a giant performance gap of 27.94% between cases where ground truth tokens are or are not captured by OCR system.

8.1.3 Annotation Bias in TextVQA

TextVQA task aims to solve the hard subset of VQA task where model is required to read the scene text from the image. However, there are 257 questions in validation datset whose ground truth answers include "not require reading text". One interesting observation is that TAP can still answer these edge cases where no text reading is required but our model tends to give out the answer of "not requires reading text". The current TextVQA evaluation metric largely depends on the human annotated ground truth as detailed in Section 6.1.

As Figure 11, the model is asked about the color of an object in the image. TAP predicts the correct color answer while our model predicts "not require reading text". Both answers make sense. However, there are 9 out of 10 ground truths voting for the color but only 1 ground truth of "not require reading text", leading to a penalization on our model's score. It is hard to fairly compare the models due to the bias in human annotation using the current evaluation. From the perspective of focusing on the TextVQA task, the evaluation metric should be slightly modified to mitigate such performance gap in "not require reading text" problems. On the other hand, using the truth answer of problems regardless of whether the text reading is required or not can improve the model's performance in a practical way and thus better integrated with the general VQA model in the future.

8.2 Future Work

8.2.1 Visual Encoder for Scene Text Reasoning

Humans can easily solve the TextVQA task by looking only at the original image. However, as mentioned in Section 7.1.1, whole image features extracted by a pretrained ViT fail to provide complementary information and only bring marginal performance when OCR tokens are absent. The main reason is that image feature extractor like ViT and ResNet are pretrained on image classificaiton tasks which does not requires scene text and their positional relations to be encoded in the features. Accordingly, our next step will be exploring the idea of substituting the information about OCR and object detection entirely with that about the whole image, and designing pre-training tasks tailored to the tasks of OCR and object detection. In this way, we will cut off the reliance of Hotpot Net on pre-trained modules, as an effort to promote model robustness. Meanwhile, with additional pre-training tasks, we expect future Hotpot Net to be capable of performing more diverse downstream tasks such as OCR and object detection other than TextVQA.

8.2.2 Data Augmentation

Image Augmentation In addition to techniques commonly used in computer vision community (rotation, scaling, adding noises, etc), which typically don't affect the semantic meaning of the visual content, VQA image augmentation also involves transformations that deliver significantly different

visual information. For instance, Gokhale et al. (2020) has proposed to mutate images via removing object instances or inverting colors, which is critical enough to lead to different answers to the questions. It is worth noting that removing object instances requires additional procedures using inpainting network based on Generative Adversarial Network (GAN) to make transformed images photorealisitic.

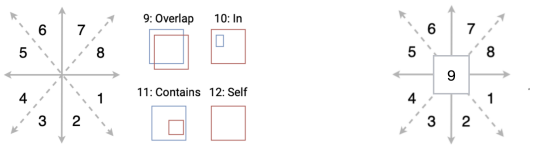


Figure 12: Positional Relationships. (left) Space of potential positional relationship between object/OCR and object/OCR. (right) Space of potential positional relationship between object/OCR and whole image.

Localization-aware QA Enrichment Visual relations are proven to be essential part in either unimodal visual tasks or multi-modal tasks. Many prior works use GAT to learn the spatial relations between objects in the images. However, such relation-aware GAT is not learned across different modalities. Though model is able to learn the spatial relationship well in visual modality, it is hard to respond to the question correctly. Inspired by the idea of implicit relation, several researchers utilizes the vanilla attention mechanism to explore the potential implicit relationship between object, OCR, and questions, but the cross-modality learning happen after the uni-modal embedding, which may compromise the cross-modality learning of relationships.

We propose a data augmentation method called Localization-aware QA Enrichment. Following SA-M4C(Kant et al., 2020), we defines 12 positional relations which can happen between `<object, object>`, `<object, OCR>`, and `<OCR, OCR>`, shown as Fig.12(left). Since we introduce whole image as a new modality, we also define 9 positional relations between `<object, IMG>` or `<OCR, iMG>`, shown as Fig.12(right). We design two templates of questions respectively:

1. Template-1: Is A [relative position] to B?

2. Template-2: Is C [relative position] to the whole image?

By adding these two types of localization-aware question-answer data, we urge the model to learn positional relationship between different modalities under the natural language supervision.

8.2.3 Out-of-domain Generalization

Apart from designing efficient and robust model structures, one of our future targets can be the generalization capabilities of Hotpot Net. Following subsections preview some potentially beneficiary techniques that we can refer to in dataset selection, model evaluation and training procedure.

Additional Dataset for Pre-training As mentioned in Yang et al. (2021), pre-training with extra data from ST-VQA, TextCaps and OCR-CC leads to an increase of up to 2.99% in validation accuracy. Therefore, our future work includes introducing additional data input for both MLM and Bounding Box Prediction tasks, and evaluating the performance improvement.

Evaluation To measure and boost model performance on Out-Of-Distribution (OOD) datasets, several papers have come up with new metrics to evaluate generalizability. Rosenberg et al. (2021) has introduced Robustness to Augmented Data (RAD), which calculates the proportion of correct answers on augmented samples among all correct answers, and has demonstrated the indicating power of RAD on model robustness against unseen data through quantitative results. Moreover, Kervadec et al. (2020) group questions on frequencies and argue that accuracy over infrequent question-answer pairs is more suitable in delineating generalization capabilites.

Training Procedure The concept of grouping questions is not merely utilized in designing evaluation metrics, but adopted in the training procedure as well. For example, Gokhale et al. (2020) has partitioned training samples by question types and optimized a different copy of the model under each cluster, which has been proved to lead to better out-of-distribution generalization.

9 Potential Ethic Issues

Since one of the major goals of TextVQA models is to serve the visually impaired, any mistake in predictions, even the tiniest one, may lead to inconvenience and even injuries. In addition, bias in our model may lead to challenges in how users perceive their surroundings and understand the world.

10 Team member contributions

Yuwei Wu contributed to the code and experiments of pre-training tasks, and composition in the sections of introduction, baselines, and proposed model.

Yanlin Feng contributed to the implementation of fine-tuning methods and their experiments, and writing task setup, analysis of fine-tuning results, limitations and future work.

Hao Wu contributed to the code and experiments of pre-training tasks, model output analysis, figures drawing, and composition in the sections of analysis, limitations, future work and ethics.

Yinghuan Zhang contributed to coding pre-training tasks, running experiments, evaluation and analysis, and writing introduction, analysis, limitations, and future work.

Jiayi Shen contributed to coding pre-training tasks, running experiments, and writing introduction, proposed model, analysis, and ethics sections.

References

Jeffrey P Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, et al. 2010. Vizwiz: nearly real-time answers to visual questions. In Proceedings of the 23nd annual ACM symposium on User interface software and technology, pages 333–342.

Ali Furkan Biten, Ron Litman, Yusheng Xie, Srikar Appalaraju, and R Manmatha. 2021. Latr: Layout-aware transformer for scene-text vqa. arXiv preprint arXiv:2112.12494.

Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marçal Rusinol, Ernest Valveny, CV Jawahar, and Dimosthenis Karatzas. 2019. Scene text visual question answering. In Proceedings of the IEEE/CVF international conference on computer vision, pages 4291–4301.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

Chenyu Gao, Qi Zhu, Peng Wang, Hui Li, Yuliang Liu, Anton Van den Hengel, and Qi Wu. 2021. Structured multimodal attentions for textvqa. IEEE Transactions on Pattern Analysis and Machine Intelligence.

Difei Gao, Ke Li, Ruiping Wang, Shiguang Shan, and Xilin Chen. 2020. Multi-modal graph neural network for joint reasoning on vision and scene text. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12746–12756.

Tejas Gokhale, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. 2020. MUTANT: A training paradigm for out-of-distribution generalization in visual question answering. CoRR, abs/2009.08566.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. arXiv preprint arXiv:1603.06393.

Wei Han, Hantao Huang, and Tao Han. 2020. Finding the evidence: Localization-aware answer prediction for text visual question answering. arXiv preprint arXiv:2010.02582.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778.

Ronghang Hu and Amanpreet Singh. 2021. Unit: Multimodal multitask learning with a unified transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1439–1449.

Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. 2020. Iterative answer prediction with pointer-augmented multimodal transformers for textvqa. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 9992–10002.

Yash Kant, Dhruv Batra, Peter Anderson, Alexander Schwing, Devi Parikh, Jiasen Lu, and Harsh Agrawal. 2020. Spatially aware multimodal transformers for textvqa. In European Conference on Computer Vision, pages 715–732. Springer.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of NAACL-HLT, pages 4171–4186.

Corentin Kervadec, Grigory Antipov, Moez Baccouche, and Christian Wolf. 2020. Roses are red, violets are blue... but should vqa expect them to? CoRR, abs/2006.05121.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In International Conference on Machine Learning, pages 5583–5594. PMLR.

Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019a. Relation-aware graph attention network for visual question answering. In Proceedings of the IEEE/CVF international conference on computer vision, pages 10313–10322.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019b. Visualbert: A simple and performant baseline for vision and language. arXiv preprint arXiv:1908.03557.

Fen Liu, Guanghui Xu, Qi Wu, Qing Du, Wei Jia, and Mingkui Tan. 2020. Cascade reasoning network for text-based visual question answering. In Proceedings of the 28th ACM International Conference on Multimedia, pages 4060–4069.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. Advances in neural information processing systems, 32.

Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 2020. 12-in-1: Multi-task vision and language representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 10437–10446.

Rafał Powalski, Łukasz Borchmann, Dawid Jurkiewicz, Tomasz Dwojak, Michał Pietruszka, and Gabriela Pałka. 2021. Going full-tilt boogie on document understanding with text-image-layout transformer. In International Conference on Document Analysis and Recognition, pages 732–747. Springer.

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. 2021. Vision transformers for dense prediction. CoRR, abs/2103.13413.

Varun Nagaraj Rao, Xingjian Zhen, Karen Hovsepian, and Mingwei Shen. 2021. A first look: Towards explainable textvqa models via visual and textual explanations. arXiv preprint arXiv:2105.02626.

Daniel Rosenberg, Itai Gat, Amir Feder, and Roi Reichart. 2021. Are VQA systems rad? measuring robustness to augmented data with focused interventions. CoRR, abs/2106.04484.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. arXiv preprint arXiv:1704.04368.

Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. 2020. Textcaps: a dataset for image captioning with reading comprehension.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 5100–5111.

Ryota Tanaka, Kyosuke Nishida, and Sen Yoshida. 2021. Visualmrc: Machine reading comprehension on document images. arXiv preprint arXiv:2101.11272.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. arXiv preprint arXiv:1710.10903.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. Advances in neural information processing systems, 28.

Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. 2021. 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.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. 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.

Zhengyuan Yang, Yijuan Lu, Jianfeng Wang, Xi Yin, Dinei Florencio, Lijuan Wang, Cha Zhang, Lei Zhang, and Jiebo Luo. 2021. Tap: Text-aware pre-training for text-vqa and text-caption. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8751–8761.

Qi Zhu, Chenyu Gao, Peng Wang, and Qi Wu. 2020. Simple is not easy: A simple strong baseline for textvqa and textcaps. arXiv preprint arXiv:2012.05153, 2.