A NOVEL TWO-STAGE MODEL WITH CROSS-LEVEL CONTRASTIVE LEARNING FOR TEXT-VQA

Jianyu Yue
Information and Communication Engineering
Harbin Engineering University
yuejy@hrbeu.edu.cn

Xiaojun Bi*, Zheng Chen
Key Laboratory of Ethnic Language Intelligent Analysis and Security Governance of MOE,
School of Information and Engineering
Minzu University of China
bixiaojun,chenzheng@hrbeu.edu.cn

ABSTRACT

Text-based Visual Question Answering (Text-VQA) task requires the model to learn effective representations in a joint semantic space. Previous methods lack the explicit alignment between object-level and scene text-level in visual-linguistic modalities. To address this issue, we propose a novel two-stage model with cross-level contrastive learning. In the first pre-training stage, we encourage the model to enhance the proximity of cross-level cross-modal representations within the same image in semantic space, while also distancing representations from different images. Then we fine-tune the model to generate the answer to the question. Experimental results on a widely used benchmark dataset demonstrate the effectiveness of our proposed model compared to existing methods.

1 INTRODUCTION

The Text-VQA task is a critical area of research, which requires the model to generate the answer to the question by thoroughly analyzing and understanding the information within a given image. For this sake, the model must possess the ability to not only recognize and comprehend the objects and scene texts within an image, but also to effectively learn their representations in a joint semantic space. Recent studies (Hu et al., 2019; Zhu et al., 2021; Gao et al., 2021; Li et al., 2023) employ different network structures to improve the performance. However, they fail to explicitly align cross-level information and generate effective joint representations, which affects the performance of the models. Inspired by contrastive learning, we propose a novel two-stage model for the Text-VQA task. In the first pre-training stage, we propose a cross-level contrastive learning method based on a multi-modal transformer model to solve the above issue, as shown in Figure 1. For the subsequent fine-tuning stage, we generate the answer in an auto-regressive manner.

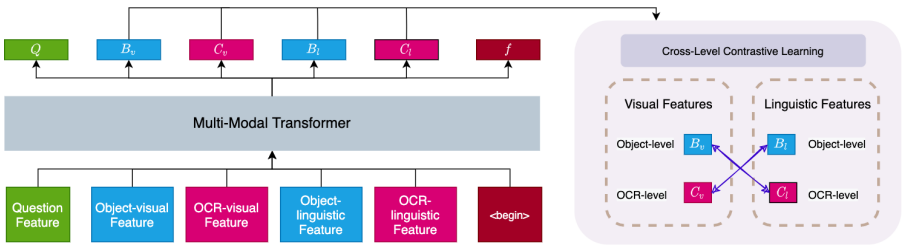


Figure 1: The proposed cross-level contrastive learning method in the pre-training process.

*Corresponding author

2 METHODOLOGY

In this section, we introduce a novel two-stage framework for the Text-VQA task, consisting of three key components. First, we extract features from the input image and question. This involves using the Faster R-CNN model (Girshick, 2015) to detect visual object regions and their category labels. We combine the appearance and location features of the objects to formulate object-level visual features. Additionally, we embed object labels by Bert (Devlin et al., 2019) to obtain object-level linguistic features. Furthermore, we employ the Microsoft Azure Optical Character Recognition (OCR) system [1] to identify scene texts in the image and extract FastText and Pyramidal Histogram of Characters (PHOC) features to generate OCR-level linguistic feature. Analogous to the object-level visual features, we extract OCR-level visual features, which comprise both appearance and spatial features.

Second, we employ a multi-modal transformer model to explicitly align the object-level and scene text-level information within the image in the pre-training stage. Inspired by contrastive learning, we design a cross-level contrastive learning (CCL) with O**b**ject-O**C**R Visual-Linguistic Contrastive **L**earning (BCL) and O**C**R-O**b**ject Visual-Linguistic Contrastive **L**earning (CBL). We design two contrastive losses for BCL and CBL by using the formulation of InfoNCE as $\mathcal{L}_{BC}$ and $\mathcal{L}_{CB}$. For each pair (visual feature v, linguistic feature l) in a batch with N pairs, we calculate the similarity between them: visual-to-linguistic as $p_n^{v,l} = \frac{exp(sim(v,l))/\tau)}{\sum_{n=1}^{N} exp(sim(v,l))/\tau)}$ and linguistic-to-visual as $p_n^{l,v} = \frac{exp(sim(l,v))/\tau)}{\sum_{n=1}^{N} exp(sim(l,v))/\tau)}$, where N is the batch size, τ is a temperature parameter, and $sim()$ denotes the cosine similarity. Then, $\mathcal{L}^{v2l}$ is defined as the cross-entropy between $p_n^{v,l}$ and the ground truth. $\mathcal{L}_{BC}$ is formulated as the mean of $\mathcal{L}_{BC}^{l2v}$ and $\mathcal{L}_{BC}^{v2l}$, utilizing object-level visual feature B_v and OCR-level linguistic feature C_l as inputs. Likewise, we could obtain the $\mathcal{L}_{CB}$.

Third, we utilize the transformer as a decoder to generate the answer to the question in the fine-tuning process. The generated answer is derived from a vocabulary list of common answers found in the training data or the OCR tokens within the image. The whole prediction process is in an auto-regressive manner.

3 EXPERIMENTAL

We conduct experiments on the TextVQA benchmark (Singh et al., 2019) by using VQA accuracy (Acc.) as the evaluation metric. Our method is compared with M4C (Hu et al., 2019), SMA (Gao et al., 2021), SSBaseline (Zhu et al., 2021), and the previous SOTA DA-Net (Li et al., 2023). As listed in Table 1, our model outperforms all considered methods. The results of our method represent a significant improvement over DA-Net, demonstrating a performance increase of 3.31% and 2.74% on the validation set and test set, respectively. This highlights the superiority of our approach.

Table 1: Comparison with the state-of-the-art methods.

Method	**Val Acc.**	**Test Acc.**
M4C (Hu et al., 2019)	39.44	39.01
SMA (Gao et al., 2021)	40.05	40.66
SSBaseline (Zhu et al., 2021)	43.95	44.72
DA-Net (Li et al., 2023)	47.12	47.11
Ours	50.43	49.85

4 CONCLUSION

In this work, we propose a novel two-stage framework with a well-designed cross-level contrastive learning for the Text-VQA task. It facilitates the alignment between object-level and scene text-level features in visual-linguistic modalities. Our methodology exhibits outstanding performance on the Text-VQA task. In the future, we plan to use this idea for the text caption task.

[1] https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision

URM STATEMENT

The authors acknowledge that at least one key author of this work meets the URM criteria of ICLR 2024 Tiny Papers Track.

REFERENCES

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019. URL https://api.semanticscholar.org/CorpusID:52967399.

Chenyu Gao, Qi Zhu, Peng Wang, Hui Li, Yuliang Liu, Anton Van den Hengel, and Qi Wu. Structured multimodal attentions for textvqa. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9603–9614, 2021.

Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. Iterative answer prediction with pointer-augmented multimodal transformers for textvqa. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9989–9999, 2019. URL https://api.semanticscholar.org/CorpusID:208006464.

Hao Li, Jinfa Huang, Peng Jin, Guoli Song, Qi Wu, and Jie Chen. Weakly-supervised 3d spatial reasoning for text-based visual question answering. *IEEE Transactions on Image Processing*, 2023.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8309–8318, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Qi Zhu, Chenyu Gao, Peng Wang, and Qi Wu. Simple is not easy: A simple strong baseline for textvqa and textcaps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 3608–3615, 2021.

A APPENDIX

A.1 IMPLEMENTATION DETAILS

We implement all models in Python using the PyTorch toolbox. We utilize the Faster R-CNN to perform object detection in the images. The visual features of each detected object consist of both appearance and spatial features. The appearance feature is generated by utilizing the Faster R-CNN fc7 weights to extract the fc6 features, resulting in a 2048-dimensional fc7 appearance feature. These fc7 weights are further fine-tuned during the training process. The 4-dimensional spatial feature comprises bounding box coordinates, which include the top-left and bottom-right points of each object. After the object label is predicted by Faster R-CNN, we input the label into Bert-base to produce a 768-dimensional object-level linguistic feature. As for the OCR-level visual feature, it comprises a similar composition with a 2048-dimensional appearance feature and a 4-dimensional spatial feature. The OCR-level linguistic feature is a blend of a 300-dimensional FastText feature and a 604-dimensional PHOC feature. The multi-modal transformer refers to the encoder-decoder architecture (Vaswani et al., 2017). In our model, we utilize a stack of 4 transformer layers, each with 12 attention heads, and the hidden dimension is set to 768. We first pre-train the model for 36,000 iterations and then fine-tune it for another 36,000 iterations. The learning rate is set to 1e-4 and the batch size is 12. For the answer vocabulary, we use the top 5000 frequent words from the answers in the training set. We set the maximum number of decoding steps to 12. In

Table 2: Ablation studies of our proposed BCL, CBL, and CCL in the pre-training stage.

Method	Val Acc.	Test Acc.
Ours	50.43	49.85
– BCL	49.28	48.56
– CBL	49.59	48.12
– CCL (BCL & CBL)	48.51	48.03

the pre-training stage, the final pre-training loss is achieved by integrating our proposed cross-level contrastive learning (CCL), masked language modeling (MLM), and image-text matching (ITM), i.e., $L_{final} = L_{CCL} + L_{MLM} + L_{ITM}$.

A.2 Details of OCR system employed

We use the Microsoft Azure OCR system to detect the texts within the image. The outputs of the Microsoft-OCR system involve three aspects: First, it efficiently extracts the textual content in the image, such as signs, labels, and written notes. Second, each piece of text identified by the system is associated with a specific visual region within the image represented by a bounding box. The bounding box is defined by the coordinates of its four points: the top-left corner, the top-right corner, the bottom-right corner, and the bottom-left corner. Third, the Microsoft-OCR system assigns a probability score to each piece of text, which indicates the system's confidence in accurately recognizing the text.

A.3 Ablation Results

We conduct ablation studies to validate the effectiveness of our proposed BCL, CBL, and the entire cross-level contrastive learning (CCL) in the pre-training stage. Table 2 presents the experimental results. It is evident that both BCL and CBL contribute to the improvement in performance. When we remove the BCL, the performance declines from 50.43% to 49.28% in the validation set and from 49.85% to 48.56% in the test set. The exclusion of CBL also leads to a decrease in performance. Removing CCL implies pre-training the model solely using MLM and ITM. The CCL brings an improvement of 1.92% (from 48.51% to 50.43%) on the validation set, it facilitates the model to learn cross-level representation in vision and linguistic modality. These results indicate that cross-level contrastive learning plays a crucial role in enhancing the performance of the model.

Table 3: Supplement analysis of components selection.

No.	CCL		Intra-Level		Intra-Modality		Val Acc.
	BCL	CBL	BBVL	CCVL	BCVV	BCLL	
1	✓	✓					50.43
2			✓	✓			49.85
3					✓	✓	49.62
4	✓	✓	✓	✓			49.55
5	✓	✓			✓	✓	49.27
6			✓	✓	✓	✓	49.38
7	✓	✓	✓	✓	✓	✓	49.47

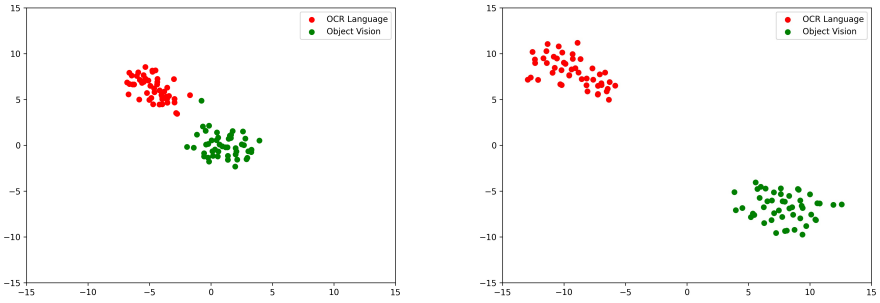
Moreover, we conduct supplementary experiments to further explore the impact of our design choices on the model's performance. Specifically, we investigate the effects of intra-level feature alignment and intra-modal feature alignment. Correspondingly, intra-level contrastive learning includes o**b**ject-level **v**isual-**l**inguistic contrastive learning (BBVL) and O**C**R-level **v**isual-**l**inguistic contrastive learning (CCVL). Intra-modal contrastive learning contains o**b**ject-O**C**R **v**isual-**v**isual contrastive learning (BCVV) and o**b**ject-O**C**R **l**inguistic-**l**inguistic contrastive learning (BCLL). Next, we explore our CCL framework with these additional contrastive learning strategies. The results of these component selection experiments are presented in Table 3. The first row, representing our proposed CCL, establishes a baseline with a validation accuracy of 50.43%. The following rows detail the results of incorporating various combinations of contrastive learning components into the framework. The introduction of BBVL and CCVL in rows 2 and 3 leads to a decline in accuracy

Figure 2: The Qualitative results showcase the comparison results between our method and the SSBaseline method in the Text-VQA task.

compared to the baseline. This decline indicates the importance of cross-level and cross-modality alignment in enhancing semantic integration within the model. Rows 4 through 7 provide a detailed exploration of various combinations of CCL with intra-level and intra-modality contrastive learning. It is worth noting that the integration of CCL with other contrastive learning methods in rows 4, 5, and 7 consistently leads to a reduction in validation accuracy. This implies that an excessive alignment of information may introduce unexpected noise and redundancy into the feature space, which could weaken the model's capacity to generate discriminative feature representations.

A.4 Qualitative results.

In Fig. 2, the samples from the TextVQA validation set showcase the comparison results intuitively in the Text-VQA task. Specifically, compared with the method SSBaseline, our approach demonstrates a significant advantage in accurately identifying key objects in question and then linking them to relevant textual components, guided by the visual attributes or labels. For instance, the second example highlights that our method enables the correct identification of the airplane as the central object and then accurately associates the question's keyword 'light blue' with the corresponding textual region in the visual context. This results in the accurate prediction of 'cargo,' as the text explicitly linked to the visual cue, instead of erroneously focusing on irrelevant background text like SSBaseline.

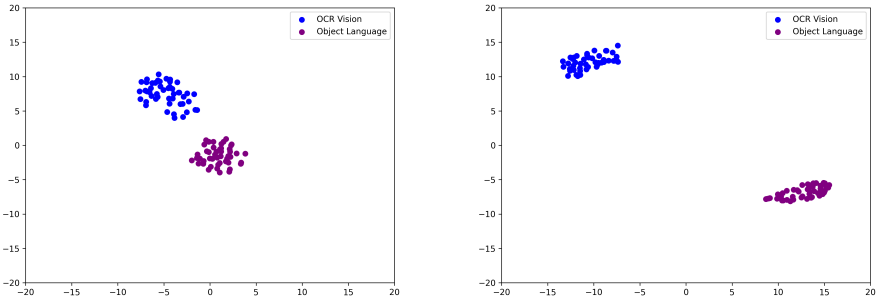


(a) Learning with BCL (b) Learning without BCL

Figure 3: The visualization for the embedding space with and w/o **Ob**ject-**OCR** visual-linguistic Contrastive **L**earning (BCL).

A.5 Visualization for the embedding space

To illustrate the effectiveness of our proposed CCL in achieving cross-level and cross-modality feature alignment, we have employed an intuitive visualization approach. This approach demonstrates



(a) Learning with CBL (b) Learning without CBL

Figure 4: The visualization for the embedding space with and w/o O**C**R-O**b**ject visual-linguistic **C**ontrastive **L**earning (CBL).

the distribution of multi-modal cross-level representations within the semantic space. Specifically, we utilize the t-SNE algorithm to transform complex three-dimensional multi-modal representations into two-dimensional feature points. When executing t-SNE, we set the number of components to 2, the perplexity to 30, the learning rate to 200, and the number of iterations to 1000 for optimal results.

In our visualization results, we compare the distribution of multi-modal feature points before and after applying our CCL, which comprises both Object-OCR Visual-Linguistic Contrastive Learning (BCL) and OCR-Object Visual-Linguistic Contrastive Learning (CBL). The results, illustrated separately in Fig. 3 and Fig. 4, demonstrate that the utilization of BCL and CBL significantly reduces the distances between aligned feature points in the semantic space. This indicates not only an improved clustering but also a more coherent feature alignment, as related points are drawn closer to one another. In contrast, without the application of CCL, the feature points exhibit greater dispersion within the semantic space.

A.6 HYPERPARAMETER ANALYSIS

Batch Size Number	**Val Acc.**
1	49.10
2	49.16
4	49.42
8	49.85
12	50.43

Table 4: Validation accuracy with respect to different batch size numbers.

Learning Rate	**Val Acc.**
5e-3	50.19
1e-4	50.43
5e-4	49.93
1e-5	49.64

Table 5: Validation accuracy with respect to different learning rates.

In contrastive learning, the temperature parameter, batch size, and learning rate are three critical hyperparameters. The temperature parameter is used to scale the similarity scores, which is vital for the model to differentiate between pairs of samples. We treat the temperature parameter as a learnable parameter, allowing the model to adaptively adjust during the training process. In our

experiments, we tune different batch size numbers (1, 2, 4, 8, 12). We can observe that as the batch size increases, the validation accuracy on the TextVQA dataset improves from 49.10% to 50.43%. This indicates that the model's performance is quite sensitive to batch size. A larger batch size provides more negative samples and facilitates representation learning. Regarding the learning rate, we explore settings including 5e-3, 1e-4, 5e-4, and 1e-5. The results show that within this range, the model is not very sensitive to changes in the learning rate. Notably, the best performance is achieved when the learning rate is set at 1e-4. Based on the aforementioned results, we select a batch size of 12 and a learning rate of 1e-4 as the final parameters.