MC-VQA using Customized Prompts

Kemmannu Vineet Venkatesh Rao
University of Michigan
kemmannu@umich.edu

Kshama Nitin Shah
University of Michigan
kshama@umich.edu

Abstract

Visual Question Answering is a challenging task and more so under the zero-shot setting. We have worked on the problem of zero-shot Visual Question Answering by leveraging a pre-trained Vision and Language model. We further propose a novel transfer methodology for the same.

Our main goal was to build a working pipeline using the tools and techniques that we learned throughout the course given the time constraints and computational constraints of **Great-lakes** The main advantage of our methodology is that it requires no training. We first generate declarative statements using the questions from the VQA-v2 dataset and use these declarative statements along with the multiple choices as the prompts to CLIP's frozen text encoder. We pass the image corresponding to that question in terms of patches and also as whole images to CLIP's frozen visual encoder. We achieve results similar to the state-of-the-art zero-shot VQA results using our methodology with fewer parameters used. Our code is available here.

1 Introduction

The ability to understand both visual and textual cues is crucial for language interactions grounded in the visual world. We want to design a system that understands the contents of an image as humans do and to be able to effectively reason about the image using natural language. Questions are a natural way for users to reason about an image. A variety of tasks in which a question is posed to an image have emerged. One of them is Visual Question Answering (VQA). Visual Question Answering is a challenging task as it requires a more deeper understanding of the visual contents of an image such as "What color is the girl's dress in the image?".

However, these Vision and Language models generally require huge amount of compute. Hence, we propose to formulate the multimodal VQA task as a downstream transfer from a large pre-trained vision and language model. Our novel pipeline thus requires lesser amount of compute and time as required by a VQA model that is trained from scratch.

Improving the performance of VQA could have far-reaching benefits in the field of personalized interaction between an AI agent and humans. Additionally, it can be used in virtual assistant systems to aid the visually impaired.

2 Related Work

Large Scale Image-Language Pre-Training of neural networks is now a popular research direction(Radford et al., 2021) (Jia et al., 2021). CLIP (Radford et al., 2021) learns joint representations of image-text paired data by pre-training a large image-text model contrastively using a visual encoder and textual encoder. These learned representations are then transferred to several downstream tasks such as Visual Question Answering, Image Classification, Object Detection, etc. CLIP (Radford et al., 2021) demonstrates superior zero-shot performance on a variety of downstream tasks such as image classification, object detection, etc. It shows the effectiveness of learning image representations using natural language as a supervisory signal.

Our methodology is inspired by recent advances in Large Language Models. Pratt *et al.*(Pratt et al., 2022) suggested that using customized prompts generated by a large language model such as GPT-3 (Brown et al., 2020) instead of the standard template-based prompts to a vision-language model for downstream tasks is more effective. In specific the CuPL (Pratt et al., 2022) generates customized prompts for each of the categories in the downstream dataset from GPT-3 using a fixed template such as "What does a(n) look like?". After generating these customized prompts, it is used to prompt the vision-language model such as CLIP for zero-

shot image classification. GLIP (Li et al., 2021) reformulates Object Detection as a grounding task by aligning each region of an image to phrases in the text prompt. GLIP is pre-trained on a large image-text dataset and can be transferred to 13 downstream object detection tasks. Our Image-Text Matching Module uses GLIP to select patches of the image that are most relevant to the given question.

Zero-shot VQA is a challenging task as it requires a more detailed understanding of the image and complex reasoning. Recent works have approached this task of zero-shot VQA in different ways. The DPT (Liu et al., 2022) involves two steps to adapt the downstream task of VQA to be similar to the pre-training task format. Firstly, it converts the question into declarative sentences and introduces the [MASK] term where the answer should be present. Secondly, it uses a task adaptation module that converts the answer prediction part to be similar to the pre-training task format of MLM (Masked Language Modeling) and ITM (Image Text Matching). We will use this declaration module to convert our questions from the VQA-v2 dataset into declarative sentences.

There are other relevant research works that have achieved notable results in the task of Zero-shot VQA. Tiong *et al.*(Tiong et al., 2022) conjoined large pre-trained language models to perform zero-shot VQA by selecting patches of the image relevant to the question and answering the question based on these selected patches. Our pipeline adopts the image-text matching module from the Plug-N-Play VQA paper to select the image patches that are relevant to each question. The QIP model (Shen et al., 2021) performs zero-shot VQA with CLIP by using prompts for the task in the standard template: "question: [question text] answer: [answer text]".

Some models have performed well in the few-shot setting and have evaluated their models in the zero-shot setting also. The TAP-C model (Song et al., 2022) converts questions from the VQA task into declarative statements and uses this as prompts to perform zero-shot VQA using the CLIP model and further performs few-shot transfer by finetuning the bias and normalization layers in the CLIP model. Our model converts questions into declarations and additionally sends relevant patches of the image to solve the zero-shot VQA task. The PICa model (Yang et al., 2021) converts images into textual descriptions that GPT-3 can understand, and then queries GPT-3 to directly predict the answer based on the question and textual descriptions generated. However, the PICa model performs a few-shot transfer whereas our pipeline performs a zero-shot transfer.

Tsimpoukelli *et al.*(Tsimpoukelli et al., 2021) attempts to ground a large language model without changing its weights, similar to prefix tuning. They transfer this captioning task to the downstream multimodal task of VQA in zero-shot and few-shot settings. This combined model shows good performance on downstream tasks such as VQA.

3 Methodology

We approach the problem of VQA using a novel pipeline (see Fig 1) that involves using a pre-trained vision-language model for the downstream multimodal task of Visual Question Answering. In our method, we used a ViT (Dosovitskiy et al., 2020) based CLIP (Radford et al., 2021) model that has been trained on more than 400 million image-text pairs. The use of CLIP-based ViT allowed us to leverage the rich representations learned by this model. Firstly, we generated customized prompts for each question and its corresponding answer instead of using the same template for all the questions. The way we did this is by using a declaration conversion module and GPT-3.

Secondly, we identified and extracted the image patches that correspond to that specific question. We did this using an Image-Text matching module. The patch extraction and selection methodology is illustrated in 2 We used these extracted patches of interest and prompts to find the similarity between them by projecting both of them into CLIP's embedding space. We then chose the prompt that ended up having the maximum similarity.

To elaborate further, for the first step, we found ways of converting a question and corresponding choices into prompts for the Vision-Language model. Doing this manually was not feasible as it required a lot of human effort and it wouldn't have been scalable for large datasets either. The next experiment we did was to leverage a pre-trained module that has been trained to convert questions into declarations. Fortunately, Liu *et al.*(Liu et al., 2022) had already trained a module to output a declaration given a question and its corresponding answers. To save on computing resources, we used their module for declaration generation. Their pre-trained

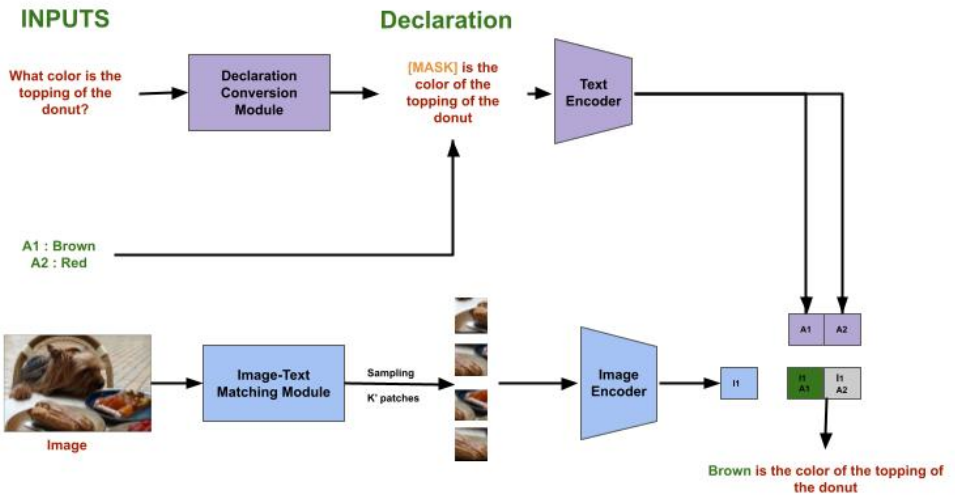

Figure 1: The system architecture of MC-VQA, consists of three main modules. Unlike previous Zero Shot VQA algorithms which input the entire image to the encoder, we sample and use relevant patches generated by our pre-trained Image-Text Matching module following the work from (Tiong et al., 2022). Further, we generate customized prompts by converting our question into a declaration using a pre-trained declaration conversion module (Liu et al., 2022) as opposed to the standard template-based prompts. Then, both the patches and prompts are used together to perform zero-shot VQA using pre-trained CLIP.

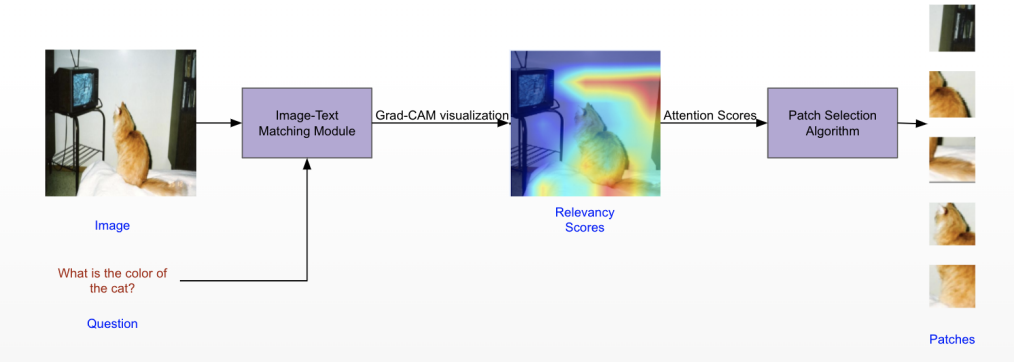

Figure 2: The architecture of our Patch Selection Pipeline , consists of the Image Text Matching Module and Patch selection algorithm. We use GradCAM (Selvaraju et al., 2019) and BLIP (Li et al., 2022) to select the image patches relevant to that particular question. Now once we have our GradCAM attention scores for each patch, we use a patch selection algorithm to sample k=20 patches with the highest attention scores outputted by GradCAM. These attention scores can be visualized as a heatmap overlayed on the original image as given in the figure

model was a T-5 transformer that was pre-trained on a question-to-declaration dataset. We passed the questions from our VQA-v2 dataset into this declaration conversion module to generate declarative statements. These declarative statements mask out the answer to the question in the declarative statement.

Another experiment that we did was to use GPT-3 to come up with these declarative statements. We initially wanted to use ChatGPT to generate these prompts however, due to an overload of requests, we decided to use GPT-3. However, we think that using ChatGPT would give us the same results if not better. We designed a specific prompt as input to GPT-3 for the three different types of questions in the VQA-v2 dataset - "other", "number" and "yes/no". To find the specific example prompts to generate declarative statements for each different question type in the dataset please refer to Appendix 7. One of the examples of the workflow of querying GPT-3 is illustrated below (see Fig 3).

Due to computational constraints, we generated declarative statements for 105,600 other type questions and 10,000 yes/no questions from the VQA-v2 dataset and evaluated our pipeline on this subset of the VQA-v2 dataset.

In our next experiment, we wanted to compare the effect of selecting image patches corresponding to a particular question versus using the entire image to answer a particular question. The intuition behind selecting patches for the zero-shot transfer was that images in the wild can be of varying resolution and in most cases, a question generally tends to focus on specific patches of the image. Hence, intuitively we expected that this method should result in higher performance. Tiong *et al.*(Tiong et al., 2022) used an *Image Matching Module* which is implemented on top of techniques like **GradCam**(Selvaraju et al., 2019) to localize a "word" in an image and then selects top k patches based on the heatmaps generated. Tiong *et al.*also suggested to randomly sample patches from those selected patches to introduce stochasticity into the process which empirically yielded better results as shown in (Tiong et al., 2022). We utilized the already implemented *Image Matching Module* from (Tiong et al., 2022) to generate patches of the images that correspond to the respective questions.

The above-described method selects small 16×16 patches corresponding to the question, calculates the alignments score by averaging the scores for each patch, and then passes the patches through CLIP's frozen image encoder, which transforms them into 224×224 size images. This may produce incorrect results because enlarging the patches does not make sense visually. Therefore, we tried a different approach to select these patches of interest. Instead of selecting the class token, we first selected patches based on their GradCam scores, then averaged the alignment scores for the patches to treat them as a new class token. Using this, we empirically yielded better results than the previously described method. This method was inspired by the Appendix section of the paper (Dosovitskiy et al., 2020) that stated that using Global Average Pooling of the features as a class token empirically yielded similar performance.

Lastly, we wanted to calculate the extent of alignment between the selected patches and the generated prompts and assign the answer as the one with the maximum alignment by averaging over all the patches. We did this in a standard **Zero Shot Transfer** methodology of converting both patches and images into the CLIP embedding space and then calculated the cosine similarity between each of the patch-prompt pairs. Our evaluation details and results are detailed in the next section.

4 Evaluation and Results

4.1 Datasets and Evaluation

We adopted the zero-shot VQA benchmark namely VQA-v2. In specific, we restricted our experiments to a subset of the validation split of the VQA-v2 dataset due to computational constraints. We used 105,600 other type questions and 10,000 yes/no type questions. We used VQA-v2 for a fair comparison with the recent works that use the VQA-v2 benchmark for the task of zero-shot VQA.

Since we reformulate the VQA task as an image classification task, we calculate the accuracy similar to an image classification task. Hence, we calculate Top-1 and TOP-5 accuracy from the predicted labels and correct answers. If the number of available options is less than 5 for a particular question, then we just calculate top-k accuracy where k is the number of the options available. Though rare, there might be a case where the correct answer is not present in the options. In this case, we add the correct answer to our list of options.

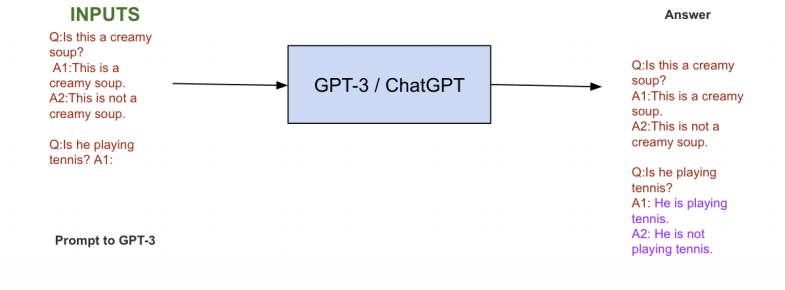

Figure 3: A specific prompt is designed to query GPT-3 with to convert questions into declarative statements. These declarative statements can then be used as customized prompts to perform zero-shot VQA

SNo.	Model	Training Details	Dataset	Question Type	Accuracy
1.	Plug N Play VQA base	Zero-Shot Transfer	VQA-v2 val	yes/no, other,number	54.3%
2.	VLKD ViT-B/16	Zero-Shot Transfer	VQA-v2 val	yes/no, other,number	38.6%
3.	Frozen VQA	Zero-Shot Transfer	VQA-v2 val	yes/no, other,number	39.1%
4.	TAP-C (w/ CLIP ViT-B/16)	Zero-Shot Transfer	VQA-v2 val	other	18.55%
5.	QIP (w/ CLIP ViT-B/16)	Zero-Shot Transfer	VQA-v2 val	other	0.70%
6.	**Our Base (whole image)**	**Zero-Shot Transfer**	**VQA-v2 val**	**yes/no,other**	**49.5%**

Table 1: Comparison Results with state of the art on zero-shot VQA.

SNo.	Visual Encoder	Top-1 Accuracy	Top-5 Accuracy
1.	ViT-B/16	23.22%	47.54%
2.	**ViT-L/14**	**23.78%**	**48.65%**
3.	ResNet-50	21.61%	45.92%

Table 2: Comparing results of effect of different Visual Encoders on zero-shot VQA. For all these experiments, we evaluated on the VQA-v2 val set and used the other type questions.

SNo.	Patch Selection Method	Top-1 Accuracy	Top-5 Accuracy
1.	using ITM module similar to (Tiong et al., 2022)	3.51%	13.28%
2.	**using GAP as a class token**	**12.61%**	**32.65%**

Table 3: Comparing results of the effect of different patch selection methods. For both these experiments, we use ViT-B/16 as the visual encoder, K=20 patches, and declarative statements as prompts and evaluated on other type questions of VQA-v2 val set. Note: GAP indicates global average pooling of ViT features

SNo.	Question type from VQA-v2 val set	Top-1 Accuracy	Top-5 Accuracy
1.	other	23.22%	47.54%
2.	yes/no	76.43%	–

Table 4: Comparing results of our best performing model (passing declarative sentences as prompts and entire images to CLIP's frozen model) on different question types of VQA-v2 validation set

4.2 Implementation Details

In our initial experiments, for the declaration conversion module, we first adopted a T-5 transformer pre-trained on a question-to-declaration dataset (Liu et al., 2022). In the later experiments, we adopted GPT-3 (Brown et al., 2020) with the **DaVinci-003** as the engine for our declaration conversion module.

To obtain the image-question matching module, we adopt BLIP (Li et al., 2022) with the ViT-L/16 architecture pre-trained on 129M image-text pairs which is adopted from (Tiong et al., 2022).

We use ViT B/16 as the visual encoder in CLIP. Hence, the CLIP model we use is CLIP ViT-B/16. However, we also performed some experiments using ViT-L/16.

Unless otherwise mentioned, we select the 8th cross-attention layer of the ITE network for Grad-CAM. We sample K=20 patches for the experiment where we use patches.

4.3 Results

For the first experiment, we used the declarative sentences generated by the pre-trained T-5 transformer and we selected patches of the image corresponding to the question using the Image-Text Matching Module. Additionally, we only use the other types of questions. We get poor results with this experiment.

For the second experiment, we used the declarative sentences generated by the pre-trained T-5 transformer and we passed the whole image to CLIP's frozen model. Additionally, we only use the other types of questions. We end up getting a 5% boost as compared to the TAP-C model (Song et al., 2022) evaluated on only other type questions from VQA-v2's validation set.

For our next experiment, we used the declarative sentences generated by the pre-trained T-5 transformer and we selected patches of the image corresponding to the question using the Image-Text Matching Module. Additionally, we global average pool the ViT's features and use this as a class token in the zero-shot evaluation instead. Further, we only use the other types of questions which are the most difficult type of questions in VQA. This gave us a 10% boost from the results we obtained in the first experiment.

Finally, we used GPT-3 to generate the declarative sentences using the yes/no questions and other types of questions. We evaluate the performance on only yes/no questions and then evaluate the performance only on other types of questions and average out the accuracy obtained from both the experiments following previous works. This model performs the best and is the final model we plan on using. We achieved comparable performance with the current SOTA model.

Comparison results with the previous SOTA and several recent models can be found in the following table (see 1). Our results show that our model's accuracy surpasses our baselines QIP and TAP-C models easily and significantly by a huge margin, thus affirming our intuition of using customized prompts for the VQA tasks. Even though our model's accuracy does not reach up to the state-of-the-art performance of Plug N Play VQA, it is relatively close to it. We would also like to stress on the fact that Plug and Play VQA uses four pre-trained large language models whereas we just use two. Hence, the FLOPS required by our model will be significantly lower than the one required by the Plug N Play VQA.

A breakdown of the performance of our best model can be found in the below table (see table 4). Our model has an accuracy of 76.43% for the yes/no questions which is expected since there exists only a binary decision for these types of questions. Our model performs significantly better than some existing models for the other type questions having a top-1 accuracy of 23.33 % and top-5 accuracy of 47.54 %.

For the Ablation studies, we wanted to stress that our method is backbone agnostic, i.e it works with both the ViT non-hierarchical-based backbones and hierarchical ResNet-based backbones. The actual performance accuracies can be seen in the following table (see table 3). Our model performs consistently for all the different backbone types. We also tested different k for selecting patches from the image and it was empirical that for k=20, the model achieved the best performance. As we feel the result of this is not relatively more important than the other results above we have decided not to include that table for different values of k in the results.

5 Conclusion

In conclusion, our experiments show that using the entire image instead of just a few selected patches when querying CLIP's frozen model leads to improved performance. This can be attributed to the

fact that using the entire image maintains the global context of the image. Additionally, we found that using global average pooling of ViT's features as a class token along with the selected patches outperforms the method where we use the ITM module following (Tiong et al., 2022). This suggests that maintaining global context and incorporating information from both the selected patches and the global features can improve performance.

6 Future Work

In the future, we hope to use the entire validation set of VQA-v2 to evaluate our pipeline. In addition to that, we plan to replace GPT-3 with ChatGPT to convert our questions into declarative sentences which could lead to better performance as ChatGPT is trained using reinforcement learning. Additionally, we hope to find more ways in which we could inject the global context of the image while passing the selected patches to CLIP's frozen model for zero-shot transfer.

7 Appendix

7.1 Some Prompt Examples for GPT-3

Prompt for yes/no type questions:
Q: Is this a creamy soup?
A1: This is not a creamy soup
A2: This is a creamy soup

Q: Is he playing tennis?
A1:

Result from GPT-3 text completion model:

Q: Is this a creamy soup?
A1: This is not a creamy soup
A2: This is a creamy soup

Q: Is he playing tennis?
A1: Yes, he is playing tennis
A2: No, he is not playing tennis

Prompt for other type and number type questions:

Q: What color are these bananas?
A: These bananas are <MASK>

Q: What is red food?
A: <MASK> is the red food.

Q: What color is the bat?
A:

Result from GPT-3 text completion model:

Q: What color are these bananas?
A: These bananas are <MASK>

Q: What is red food?
A: <MASK> is the red food.

Q: What color is the bat?
A: The bat is <MASK>

Note: the portion that is highlighted is what was generated by GPT-3

7.2 Visualizations

In the appendix, we show visualizations of GradCAM heatmaps and the generated declarative statements (prompts) for VQA-v2 in the following page.

These examples from VQA-v2 dataset are from the subset of other type questions. The visualizations for yes/no questions would be the same since it's a relatively easier task. Hence, we have decided not to include it in the report to manage space. We show how patches can get selected based on the GradCAM heatmap.

From the 4, we can see that GradCAM does not always localize the object of interest very well as shown in some of the examples above. In the example of the question - "Is the person a male or female?", GradCAM isn't able to effectively localize the person in the image. Better alternatives to GradCam such as GradCam+, EigenCam, and ScoreCam might lead to better results, though experiments in this are left as future work.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An image is worth 16x16 words: Transformers for image recognition at scale.

Figure 4: Some examples from VQA-v2 dataset from the subset of other type questions. The visualizations for yes/no questions would be the same since it's a relatively easier task we have decided not to include it in the report for space management. We show how patches can get selected based on the GradCAM heatmap. We can also see that GradCAM does not always localize the object of interest very well as shown in some of the examples above.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. *CoRR*, abs/2102.05918.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. 2022. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. *CoRR*, abs/2201.12086.

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. 2021. Grounded language-image pre-training. *CoRR*, abs/2112.03857.

Yuhang Liu, Wei Wei, Daowan Peng, and Feida Zhu. 2022. Declaration-based prompt tuning for visual question answering.

Sarah Pratt, Rosanne Liu, and Ali Farhadi. 2022. What does a platypus look like? generating customized prompts for zero-shot image classification.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2019. Grad-CAM: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359.

Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. 2021. How much can CLIP benefit vision-and-language tasks? *CoRR*, abs/2107.06383.

Haoyu Song, Li Dong, Wei-Nan Zhang, Ting Liu, and Furu Wei. 2022. Clip models are few-shot learners: Empirical studies on vqa and visual entailment.

Anthony Meng Huat Tiong, Junnan Li, Boyang Li, Silvio Savarese, and Steven C. H. Hoi. 2022. Plug-and-play vqa: Zero-shot vqa by conjoining large pre-trained models with zero training.

Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *CoRR*, abs/2106.13884.

Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. 2021. An empirical study of GPT-3 for few-shot knowledge-based VQA. *CoRR*, abs/2109.05014.