SOrT-ing VQA Models : Contrastive Gradient Learning for Improved Consistency

Sameer Dharur[1] **Purva Tendulkar**[1→3] **Dhruv Batra**[1,2] **Devi Parikh**[1,2] **Ramprasaath R. Selvaraju**[1→4]
[1]Georgia Institute of Technology [2]Facebook AI Research
[3] University of California, San Diego [4] Salesforce Research
{sameerdharur, purva, dbatra, parikh, ramprs}@gatech.edu

Abstract

Recent research in Visual Question Answering (VQA) has revealed state-of-the-art models to be inconsistent in their understanding of the world – they answer seemingly difficult questions requiring reasoning correctly but get simpler associated sub-questions wrong. These sub-questions pertain to lower level visual concepts in the image that models ideally should understand to be able to answer the higher level question correctly. To address this, we first present a gradient-based interpretability approach to determine the questions most strongly correlated with the reasoning question on an image. We use this to evaluate VQA models on their ability to identify the relevant sub-questions needed to answer a reasoning question. Next, we propose a contrastive gradient learning based approach called Sub-question Oriented Tuning (SOrT) which encourages models to rank relevant sub-questions higher than irrelevant questions for an <image, reasoning-question> pair. We show that SOrT improves model consistency by upto 6.5% (absolute) over existing baselines, while also improving visual grounding.

1 Introduction

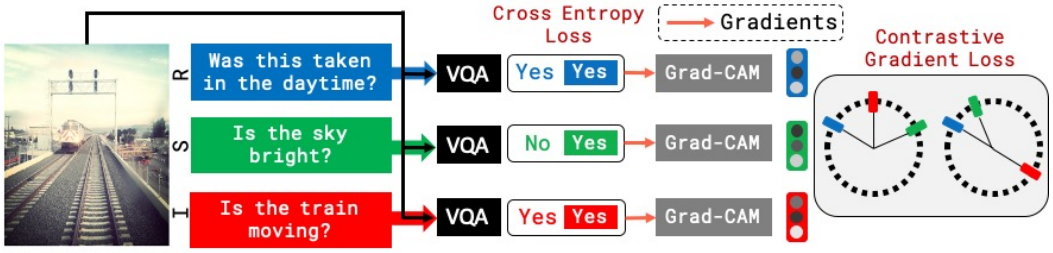
Current Visual Question Answering (VQA) models have problems with consistency. They often correctly answer complex reasoning questions, i.e, those requiring common sense knowledge and/or logic on top of perceptual capabilities (e.g, "Was this taken in the daytime?" in Fig 1), but fail on associated low level perception questions, i.e., those directly related to the visual content in the image (e.g, "Is the sky bright?" in Fig 1). This indicates that the models likely answered the reasoning question correctly for the wrong reason(s). In this work, we explore the usefulness of leveraging information about *sub-questions*, i.e., low level perception questions related to a reasoning question, and *irrelevant questions*, i.e., any other questions about the image that are unrelated to the reasoning question, to improve consistency in VQA models.

[12] studied this problem and introduced the VQA-Introspect dataset that draws a distinction between higher-level reasoning questions and lower-level perception sub-questions. We augment this dataset with additional perception questions from the VQAv2 dataset such that each <image, reasoning question> pair contains a set of *sub-questions* (e.g.,"Is the sky bright?" in Fig 1) and *irrelevant questions*(e.g., "Is the train moving?" in Fig 1).

We use Gradient-based Class Activation Mapping (Grad-CAM) vectors [10] – a faithful function of the model's parameters, question, answer and image – to interpret the questions most strongly correlated with a reasoning question for a model. This is measured by ranking questions based on the cosine similarity of their Grad-CAM vectors with that of the reasoning question. We find that even top-performing VQA models often rank *irrelevant questions* higher than the relevant *sub-questions*.

1st NeurIPS workshop on Interpretable Inductive Biases and Physically Structured Learning (2020), virtual.

Figure 1: The approach for SOrT. The reasoning question *Was this taken in the daytime?* has the sub-question *Is the sky bright?* and an irrelevant question *Is the train moving?* We tune the model using cross entropy losses and a contrastive gradient loss to bring the reasoning question's Grad-CAM vector closer to that of its sub-question, and distance it from that of its irrelevant question.



To correct this, we introduce a new contrastive gradient learning approach that fine-tunes a VQA model by enforcing *sub-questions* to be ranked higher than *irrelevant questions* while answering a reasoning question. This is achieved by forcing the cosine similarity of the reasoning question's Grad-CAM vector with that of a *sub-question* to be higher than with that of an *irrelevant question*. Our approach improves the model's consistency, defined as the frequency with which the model correctly answers a *sub-question* given that it correctly answers the reasoning question.

Additionally, we assess the effects of our approach on visual grounding by comparing Grad-CAM heatmaps with human attention maps collected in the VQA-HAT dataset [3]. We find that our approach of enforcing this better ranking of *sub-questions* also improves visual grounding.

2 Related Work

Visual Question Answering. The VQA task [2] requires answering a free form natural language question about visual content in an image. Previous work has shown that models often perform well on the task by exploiting language and dataset biases [1, 14]. To evaluate if these models are consistent in reasoning, [12] introduced the VQA-Introspect dataset, containing human explanations in the form of sub-questions and answers for questions in the VQA dataset requiring higher level reasoning.

Model Interpretability. While prior work has attempted to explain VQA decisions in the visual modality [10, 11, 9], the multi-modal task of VQA has a language component which cannot always be explained visually, i.e., visual regions can be insufficient to express underlying concepts [5, 6]. [8] and [13] generate textual justifications through datasets curated with human explanations. Our approach differs by being fully self-contained and faithful to the model.

Aligning network importances. [11] introduced an approach to align visual explanations with regions deemed important by humans, thereby improving visual grounding in VQA models. In followup work, [12] explored aligning attention maps for the reasoning question and associated *sub-questions* from VQA-Introspect to improve language based grounding. In contrast to attention maps, our work encourages Grad-CAM vectors of a reasoning question to align with those of *sub-questions* and disalign with those of *irrelevant questions*. Intuitively, this means that we are making the neurons used while answering a reasoning question to be similar to those used while answering a *sub-question* and dissimilar to those used while answering an *irrelevant question*. Our experiments show that this approach improves the model's consistency and visual grounding.

3 Approach

3.1 Preliminaries

Grad-CAM. Grad-CAM was introduced in [10] as a technique to obtain visual explanations from any CNN-based deep network. In this work, we adopt Grad-CAM to compute the contribution of a neuron at the layer in a VQA model where the vision and language modalities are combined. This is computed by taking the gradient of the predicted output class score with respect to the neuron

activations in the layer, and then point-wise multiplying it with the activations. Specifically, if y^c denotes the score of the ground-truth output class and A_k the activations of layer k of the model, the Grad-CAM importance vector G_k^c (or simply, Grad-CAM vector) is computed as,

$$G_k^c = \frac{\partial y^c}{\partial A_k} * A_k \tag{1}$$

Unlike Grad-CAM visualizations, these vectors are not visually interpretable as they are not computed on the final convolutional layer of the CNN.

Dataset. We construct our dataset by augmenting VQA-Introspect with perceptual question-answer pairs from VQAv2 [4]. The training/val splits contain 54,345/20,256 <image, reasoning question> pairs with an average of 2.58/2.81 *sub-questions* and 7.63/5.80 *irrelevant questions* for each pair.

3.2 Sub-question Oriented Tuning

The key idea behind Sub-question Oriented Tuning (SOrT) is to encourage the neurons most strongly relied on (as assessed by Grad-CAM vectors) while answering a reasoning question ("Was this taken in the daytime?" in Fig 1) to be similar to those used while answering the *sub-questions* ("Is the sky bright?") and dissimilar to those used while answering the *irrelevant questions* ("Is the train moving?"). This enforces the model to use the same visual and lingustic concepts while making predictions on the reasoning question and the *sub-questions*. Our loss has the following components.

Contrastive Gradient Loss. With the Grad-CAM vectors of the reasoning question (G_R), *sub-question* (G_S) and *irrelevant question* (G_I), our contrastive gradient loss $\mathcal{L}_{\text{CG}}$ is,

$$\mathcal{L}_{\text{CG}} = \max\left(0, \overbrace{\frac{G_R \cdot G_I}{|G_R||G_I|}}^{\text{cosine-sim}(G_R, G_I)} - \underbrace{\frac{G_R \cdot G_S}{|G_R||G_S|}}_{\text{cosine-sim}(G_R, G_S)}\right) \tag{2}$$

Binary Cross Entropy Loss. To retain performance of the model on the base task of answering all questions correctly, we add a Binary Cross Entropy Loss term ($\mathcal{L}_{\text{BCE}}$) to penalize incorrect answers.

Total Loss. Let o_R, gt_R, o_S, gt_S, $\text{o}_\text{S'}$ and $\text{gt}_\text{S'}$ represent the predicted and ground-truth answers for the reasoning, *sub-questions* and *irrelevant questions* respectively, and λ_1, λ_2, λ_3 be tunable hyper-parameters. Our total loss $\mathcal{L}_{\text{SOrT}}$ is,

$$\mathcal{L}_{\text{SOrT}} = \mathcal{L}_{\text{CG}} + \lambda_1 \mathcal{L}_{\text{BCE}}(\text{o}_\text{R}, \text{gt}_\text{R}) + \lambda_2 \mathcal{L}_{\text{BCE}}(\text{o}_\text{S}, \text{gt}_\text{S}) + \lambda_3 \mathcal{L}_{\text{BCE}}(\text{o}_\text{I}, \text{gt}_\text{I}) \tag{3}$$

4 Experiments

Baselines. We compare SOrT against the following baselines: **1) Pythia** [7], and **2) SQuINT** in which, as discussed in Sec 2, [12] fine-tuned Pythia with an attention alignment loss to ensure that the model looks at the same regions when answering the reasoning and *sub-questions*.

4.1 Metrics

Ranking. 1) Mean Precision@1 (MP@1). Proportion of <image, reasoning question> pairs for which the highest ranked question is a *sub-question*. **2) Ranking Accuracy**. Proportion of <image, reasoning question> pairs whose *sub-questions* are all ranked above their *irrelevant questions*. **3) Mean Reciprocal Rank (MRR)**. Average value of the highest reciprocal rank of a *sub-question* among all the <image, reasoning question> pairs. Higher is better. **4) Weighted Pairwise Rank (WPR) Loss**. This searches for pairs of incorrectly ranked <sub, irrelevant> questions and computes the differences of their similarity scores with the reasoning question. Averaged across all pairs, this computes the *extent* by which rankings are incorrect. Lower is better.

Model Performance. 1) Quadrant Analysis. a. **R✓ S✓** The pairs where reasoning and *sub-questions* are both correctly answered. b. **R✓ S✗** The pairs where reasoning question is correctly

Table 1: Results on the Consistency, Accuracy and Ranking metrics described in Sec 4.1. Consistency and Ranking are benchmarked on VQA-Introspect's val split, VQA Accuracy on VQAv2's val split.

	Consistency Metrics					Accuracy Metrics		Ranking Metrics			
Method	R✓ S✓ ↑	R✓ S✗ ↓	R✗ S✓ ↓	R✗ S✗ ↓	Consistency% ↑	Reas. Accuracy% ↑	VQA Accuracy% ↑	MP@1 ↑	Ranking Accuracy ↑	MRR ↑	WPR ↓
Pythia	50.61	19.88	**17.15**	12.36	71.81	**75.15**	**64.95**	57.75	30.33	71.87	52.75
Pythia + SQuINT	53.03	17.58	18.63	10.74	75.10	74.95	64.75	55.87	29.45	71.49	**39.20**
Pythia + SOrT	**54.62**	**15.09**	20.31	**9.97**	**78.35**	74.18	64.07	**61.73**	**31.90**	**74.43**	40.03

answered, while the *sub-question* is incorrectly answered. c. **R✗ S✓** The pairs where reasoning question is incorrectly answered, while the *sub-question* is correctly answered. d. **R✗ S✗** The pairs where reasoning and *sub-questions* are both incorrectly answered. **2) Consistency.** The frequency with which a model correctly answers a *sub-question* given that it correctly answers the reasoning question, i.e, **R✓ S✓** /(**R✓ S✓** + **R✓ S✗**). **3) Reasoning Accuracy**. The accuracy on the reasoning split of VQAv2 dataset, and **4) Overall Accuracy**. Accuracy on the VQAv2 validation set. More details on the metrics are in the Appendix.

4.2 Results

Does SOrT better identify perception questions relevant for answering a reasoning question? As described in Sec 3.2, the model ranks perception questions (*sub-questions* and *irrelevant questions*) associated with an <image, reasoning question> pair according to the cosine similarities of their Grad-CAM vectors with that of the reasoning question. As seen in Table 1, we find that our approach outperforms both its baselines on nearly all the ranking metrics. We observe gains of 4-6% on MP@1 and MRR, and 1.5-2.5% on Ranking Accuracy. Likewise, the improvement in WPR - the soft metric that computes the *extent* by which rankings are incorrect - is a major 12% over Pythia. Our approach thus better distinguishes relevant and irrelevant perceptual concepts for a reasoning question.

Does recognizing relevant sub-questions make models more consistent? We find that the improved ranking of *sub-questions* through SOrT improves consistency by 6.5% points over Pythia and 3.25% points over SQuINT[1] (a qualitative example can be found in the Appendix). As seen in Table 1, the consistency gains are due to significant improvements in the R✓ S✓ and R✓ S✗ quadrants. This, however, comes at the expense of a drop in accuracies by $\sim$1% point.

Gradient-based explanations have been shown to be more faithful to model decisions compared to attention maps [11]. Our results confirm this by showing that aligning Grad-CAM vectors makes models more consistent compared to aligning attention maps, as done in SQuINT.

Does enforcing language-based alignment lead to better visual grounding? To evaluate this, we compute visual grounding through Grad-CAM applied on the final convolutional layer. We then compute the correlation of Grad-CAM heatmaps with VQA-HAT [3]. We find that our SOrT gets a Spearman rank correlation of 0.103 ± 0.008 compared to 0.080 ± 0.009 for Pythia and 0.060 ± 0.008 for SQuINT. These improvements indicate that enforcing language-based alignment during training improves visual grounding on an unseen validation split. A qualitative example of this improved visual grounding is included in the Appendix.

5 Discussion

In this work, we developed language-based interpretability metrics that measure the relevance of a lower-level perception question for answering a higher-level reasoning question. Evaluating state-of-the-art VQA models on these metrics reveals that these models often rank *irrelevant questions* higher than relevant *sub-questions*. To address this, we present Sub-question Oriented Tuning (SOrT), a contrastive gradient learning approach for teaching VQA models to distinguish between relevant and irrelevant perceptual concepts while answering a complex reasoning question. This is done by appropriately aligning the corresponding Grad-CAM vectors. We demonstrate SOrT's effectiveness at making VQA models more consistent without significantly affecting their overall predictive performance. We also show that this alignment achieves better visual grounding.

[1]These numbers are averaged values from 10-fold cross validation runs on the val split. The std dev values observed were 0.3 for Pythia and 0.41 for SQuINT and SOrT.

References

[1] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering, 2017.

[2] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, and Devi Parikh. Vqa: Visual question answering, 2015.

[3] Abhishek Das, Harsh Agrawal, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. Human Attention in Visual Question Answering: Do Humans and Deep Networks Look at the Same Regions? In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

[4] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering, 2016.

[5] Yash Goyal, Akrit Mohapatra, Devi Parikh, and Dhruv Batra. Towards transparent ai systems: Interpreting visual question answering models, 2016.

[6] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to reason: End-to-end module networks for visual question answering. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[7] Yu Jiang, Vivek Natarajan, Xinlei Chen, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Pythia v0.1: the winning entry to the vqa challenge 2018, 2018.

[8] Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. Multimodal explanations: Justifying decisions and pointing to the evidence, 2018.

[9] Tingting Qiao, Jianfeng Dong, and Duanqing Xu. Exploring human-like attention supervision in visual question answering, 2017.

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

[11] Ramprasaath R. Selvaraju, Stefan Lee, Yilin Shen, Hongxia Jin, Shalini Ghosh, Larry Heck, Dhruv Batra, and Devi Parikh. Taking a hint: Leveraging explanations to make vision and language models more grounded, 2019.

[12] Ramprasaath R. Selvaraju, Purva Tendulkar, Devi Parikh, Eric Horvitz, Marco Ribeiro, Besmira Nushi, and Ece Kamar. Squinting at vqa models: Introspecting vqa models with sub-questions, 2020.

[13] Jialin Wu and Raymond J. Mooney. Faithful multimodal explanation for visual question answering. In *Proceedings of the Second BlackboxNLP Workshop at ACL*, pages 103–112, Florence, Italy, August 2019.

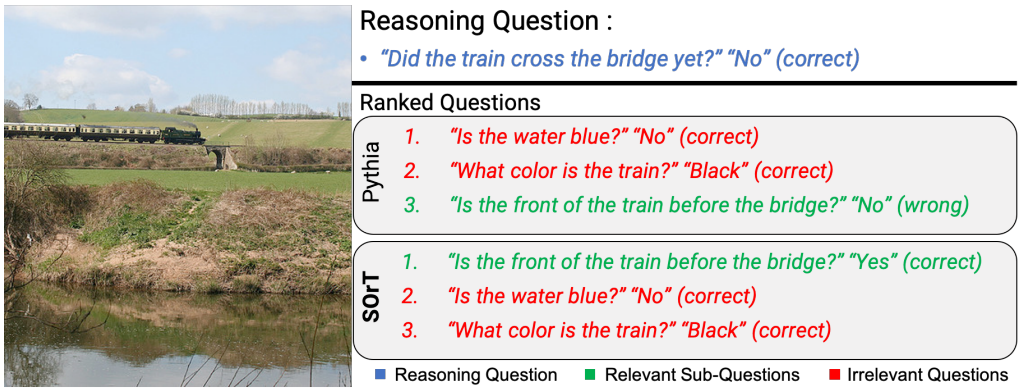
[14] Peng Zhang, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Yin and yang: Balancing and answering binary visual questions, 2015.

6 Appendices

6.1 Model Consistency

In Sec 4.2, we described the improvement in consistency arising from our approach of SOrT-ing VQA models. Fig 2 shows a qualitative example of this. The Pythia model answers its *sub-question* incorrectly. After SOrTing, our model now ranks the relevant *sub-question* higher than the *irrelevant* ones and also answers it correctly – thus improving consistency.

Figure 2: An example of improvement in consistency between Pythia (top) and SOrT (below) brought about by better sub-question ranking.



6.2 Visual Grounding

In Sec 4.2, we quantify the gains in visual grounding as measured by computing Spearman rank correlation between Grad-CAM heatmaps and human attention maps. The example in Fig 3 demonstrates the superior grounding of SOrT compared to its baselines. For the question "*Is the baby using the computer?*" and its corresponding answer "*Yes*", we see that the Grad-CAM heatmap generated by SOrT is closest to that of the human attention map. It is also the only heatmap in this example that actually points to the fingers of the child which is the essential visual component for answering the question.

Figure 3: A qualitative example of the improvement in visual grounding by SOrT. For the <question, answer> pair of <*"Is the baby using the computer?", "Yes"*>, we see the comparison of the Grad-CAM heatmaps generated by the 3 models and the human attention map. SOrT's heatmap is most closely aligned with that of the human attention map.

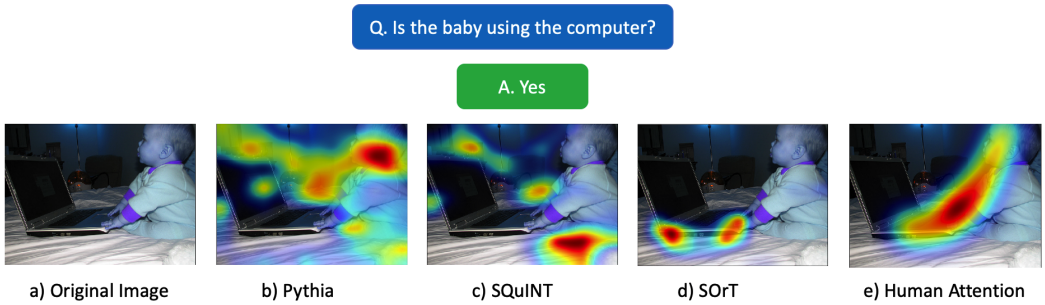
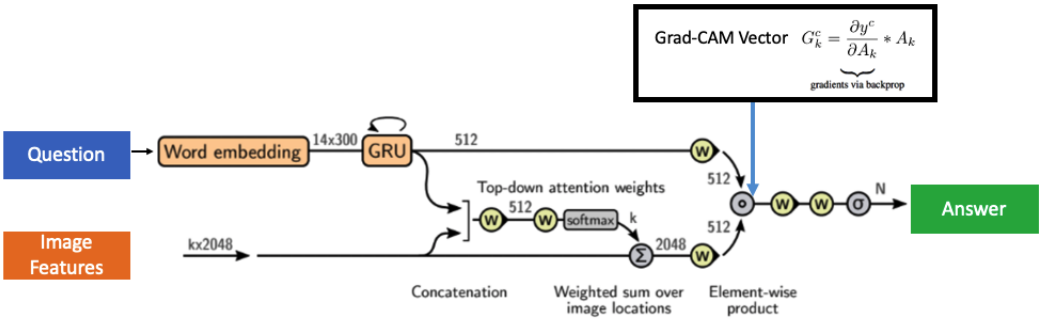


Figure 4: The architecture of the Pythia model.



6.3 Experimental Details

6.3.1 Algorithms

We use the Pythia model (Fig 4) for our experiments. Specifically, for our SOrT approach, we compute Grad-CAM vectors for the reasoning question, *sub-questions* and *irrelevant questions* on each image at the layer where the vision and language modalities are combined. We then use customized losses described in Sec 3.2 of the paper. The mathematical computation of consistency is described in Sec 4.1, while the ranking metrics are described below.

Mean Precision@1 (MP@1). For a given ordering of related questions (based on 1 of the 3 similarity scores), we compute the fraction of pairs in which a relevant perception sub-question was ranked the highest, i.e, had the highest similarity score with that of the reasoning question. This is equivalent to setting a bare-bones expectation of reasoning ability for the model - "Among all the related questions for a pair, was atleast the highest ranked related question a relevant perception sub-question?"

This is illustrated in an example below across two sets.

Example Query 1 : "What is the capital of the USA?"

Predicted Ranking 1 : ["New York", "Washington DC", "San Francisco"]

Ground Truth Answers 1 : [0, 1, 0]

Example Query 2 : "Where is the Golden Gate Bridge located?"

Predicted Ranking 2 : ["San Francisco", "Atlanta", "Los Angeles"]

Ground Truth Answers 2 : [1, 0, 0]

Across these two examples, the Mean Precision@1 value would be $\frac{1}{2}$ since only one of them has its highest ranked item as a correct answer.

Ranking Accuracy. This computes the proportion of pairs in which all the relevant perception sub-questions are ranked higher than the irrelevant questions. This would represent a perfect ranking capability of the model.

Example Query 1 : "Cities in Asia."

Predicted Ranking 1 : ["Stockholm", "Beijing", "New Delhi"]

Ground Truth Answers 1 : [0, 1, 1]

Example Query 2 : "Planets in the solar system."

Predicted Ranking 2 : ["Neptune", "Jupiter", "Phobos"]

Ground Truth Answers 2 : [1, 1, 0]

The combined Ranking Accuracy across these two examples would be $\frac{1}{2}$ since all the correct answers are ranked higher than the incorrect ones only in the second set.

Mean Reciprocal Rate (MRR). This is a variation of MP@1 which captures the highest rank of a relevant item in a list. In our case, the reciprocal rank is concerned with the highest rank of a relevant perception sub-question among all the ranked related questions for a pair. The reciprocal of this highest relevant rank is averaged across the entire dataset. This is represented in the example below.

Example Query 1 : “What is the capital of the USA?"

Predicted Ranking 1 : [“New York", “Washington DC", “San Francisco"]

Ground Truth Answers 1 : [0, 1, 0]

Example Query 2 : “Where is the Golden Gate Bridge located?"

Predicted Ranking 2 : [“San Francisco", “Atlanta", “Los Angeles"]

Ground Truth Answers 2 : [1, 0, 0]

Across these two examples, the MRR could be calculated as follows :

$$\begin{aligned} RR_1 &= \frac{1}{2} \\ RR_2 &= \frac{1}{1} = 1. \\ MRR &= \frac{1}{2} * (RR_1 + RR_2) = \frac{1}{2} * \frac{3}{2} = \frac{3}{4} \end{aligned} \tag{4}$$

Weight Pairwise Rank (WPR) Loss. All the above metrics only account for the ranking of the candidate questions for a given pair, but do not consider the *extent* by which these questions differ in their rankings. Concretely, to have a comprehensive understanding of the relevance of each question, we need to account for the magnitude of their similarity scores with the reasoning question in our overall metric.

For a pair, we create a parallel list of ranked questions in which all the relevant perception sub-questions are higher than the other questions, while retaining the same similarity scores as computed for the originally ranked list. We then compare these two lists pair-wise, i.e, in each index, and sum up the differences of the similarity scores if the rankings are different between the two lists. This provides us a way to measure not just the deviation from the desired order of rankings but also the magnitude of the differences in similarity scores which are responsible for the erroneous rankings. If S could be represented as the set of size n containing all such incorrectly ranked pairs $(r.r')$ with scores (α, α'), we could compute the WPR loss for each set as the sum of the absolute values of the differences between each α and α'.

$$\text{WPR} = \frac{\sum_{(r,r') \in S} |\alpha - \alpha'|}{n} \tag{5}$$

This is then averaged across the entire dataset.

We illustrate an example for a single set.

Query : "Which of these is a national capital?"

Predicted Ranking With Scores : [(“Mexico City", 0.9), (“Miami", 0.8), (“Copenhagen", 0.7)]

Ground Truth Answers : [1, 0, 1]

Parallel List w.r.t Ground Truth Answers : [(“Mexico City", 0.9), (“Copenhagen", 0.7), (“Miami", 0.8)]

$$\text{WPR} = \frac{1}{2} * (0.1 + 0.1) = 0.1 \tag{6}$$

6.3.2 Source Code

Our source code is publicly accessible at https://github.com/sameerdharur/sorting-vqa.

6.3.3 Computing Infrastructure

The computing infrastructure used for training and running the models described in the paper was 1 NVIDIA TITAN Xp GPU.

6.3.4 Runtime

The average training time for the model on each combination of hyperparameters was roughly 12 hours.

6.3.5 Parameters

The details on the parameters of the model can be found in the Pythia paper referenced in the main section.

6.3.6 Validation Performance

The results of the validation performance on each of the different metrics have been included in Sec 4.2 of the main section. The metrics have been explained above, with the source code linked above.

6.3.7 Hyperparameter Search

For the best performing models, the values of λ described in the losses of Sec 3.2 are $\lambda_1 = \lambda_2 = 2.27, \lambda_3 = 0.0003$. These values were selected based on the differing scales of the loss components and chosen from running hyperparameter sweeps. The rest of the hyperparameters were unchanged from those reported for the best performing Pythia model.

A total of 294 hyperparameter trial runs were conducted with λ_1 and λ_2 ranging from 0.025 to 25, and λ_3 ranging from $1e-5$ to 100.

These values were picked by a combination of uniform sampling and random tuning, and were optimized on a combination of consistency and accuracy. As mentioned in Sec 4.2, the expected validation results fall within the statistical range of the results defined by a standard deviation of 0.3 and 0.41 for Pythia and SQuINT/SOrT.

6.3.8 Datasets

As detailed in Sec 3.1, our dataset is a combination of the VQA-Introspect and VQAv2 datasets. In total, our train/val splits contain 54,345/20,256 <image, reasoning question> pairs with an average of 2.58/2.81 sub-questions and 7.63/5.80 irrelevant questions for each pair respectively. Subsets of this data have been attached in a zip file with this submission to serve as representative examples. The details on the VQA-Introspect and VQAv2 datasets, which are publicly available, can be found in the corresponding papers cited in the main section. The VQA-HAT dataset used in the visual grounding analysis is also publicly accessible at this link.