COL774: MACHINE LEARNING
ASSIGNMENT 4: VISUAL QUESTION ANSWERING

Avni Aggarwal 2022CS11605
Aniket Chattopadhyay 2022CS11599

Introduction

Visual Question Answering (VQA) is a multimodal task that lies at the intersection of computer vision and natural language processing. Given an image and a related textual question, the goal is to predict a textual answer. This task poses significant challenges as it demands joint understanding of visual scenes and natural language, often requiring reasoning, object recognition, counting, and spatial understanding.

In this assignment, we tackle the VQA problem using the CLEVR dataset—a synthetic dataset specifically designed to evaluate compositional language and visual reasoning. It consists of scenes containing multiple 3D objects with varying attributes (color, shape, size, material), accompanied by diverse and challenging question-answer pairs.

Implementation Details

To effectively address this task, we implement an end-to-end trainable multi-modal Transformer-based model. The architecture consists of four major components:

- **Image Encoder:** A pretrained ResNet101 model is used to extract spatial feature maps from images. The final fully connected and pooling layers are removed to retain spatial structure in the visual features.
- **Language Encoder:** A Transformer model (using a pre-trained BERT tokenizer) encodes the input question into a dense sequence of feature representations.
- **Feature Fusion Module:** Cross-attention mechanisms are used to combine the language and vision modalities into a single joint representation that captures the interactions between image regions and question semantics.
- **Decoder:** A feedforward classifier head that maps the fused representation to one of the possible answer classes.

The entire model is trained in an end-to-end fashion, meaning the weights of both the image encoder and the language encoder (as applicable) are updated during training. Below are the key implementation details common across all parts:

- Optimizer: Adam
- Loss Function: Cross-Entropy Loss

- Learning Rate: 1e-5
- Batch Size: 32
- Tokenization: BERT tokenizer (bert-base-uncased)
- Image Backbone: ResNet101 (pretrained on ImageNet)

By following this architecture and training scheme, the model is equipped to learn complex interactions between the visual and textual modalities.

Part 8

- In this part, we had to use ResNet101 purely for feature extraction, we keep its parameters frozen during training by setting resnet.requires grad = False.
- Stopping criteria is set such that if the difference in validation loss between the current epoch and the previous epoch is less than a threshold (0.001) for 5 continuous epochs, early stopping is triggered.
- Using this technique, training completes at 18 epochs.

Epoch	**Train Loss**	**Train Accuracy**	**Val Loss**	**Val Accuracy**
1	1.0829	0.5036	0.9832	0.5234
2	0.9135	0.5636	0.9027	0.5749
3	0.8643	0.5905	0.9008	0.5921
4	0.8320	0.6075	0.8517	0.6059
5	0.7998	0.6261	0.8266	0.6263
6	0.7649	0.6476	0.7981	0.6425
7	0.7330	0.6663	0.7942	0.6522
8	0.7054	0.6799	0.7760	0.6621
9	0.6824	0.6916	0.7489	0.6650
10	0.6595	0.7030	0.7615	0.6720
11	0.6394	0.7130	0.7449	0.6734
12	0.6248	0.7189	0.7691	0.6794
13	0.6108	0.7255	0.7409	0.6816
14	0.5973	0.7328	0.7469	0.6838
15	0.5839	0.7389	0.7580	0.6827
16	0.5710	0.7454	0.7634	0.6841
17	0.5341	0.7632	0.7753	0.6854
18	0.5197	0.7697	0.7901	0.6861

Table 1: Train and Validation metrics - Part 8

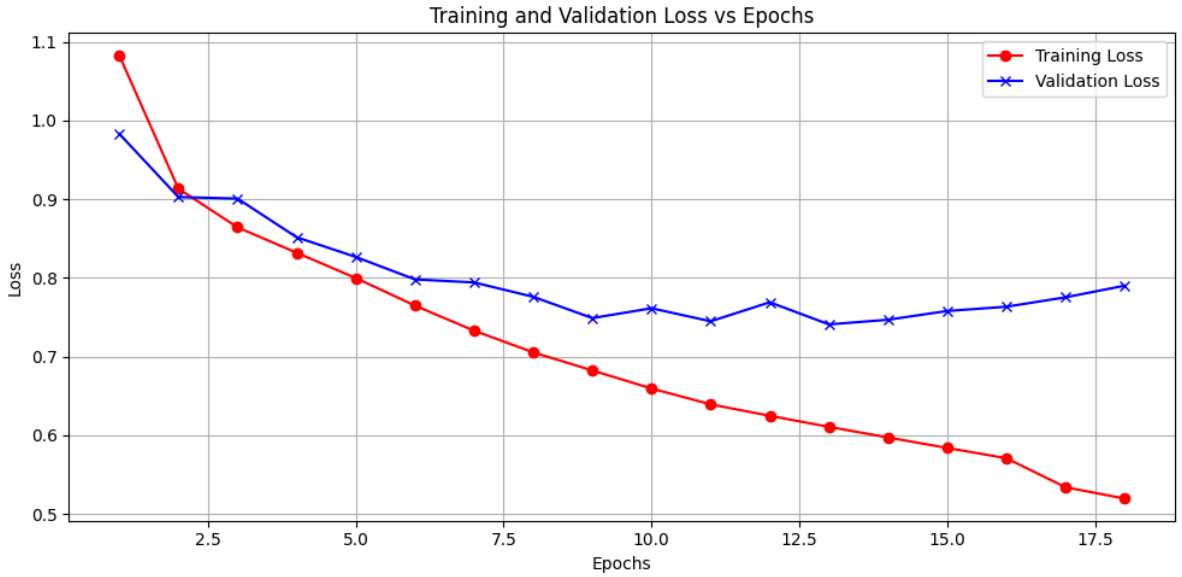


Figure 1: Training and Validation Loss vs Number of Epochs

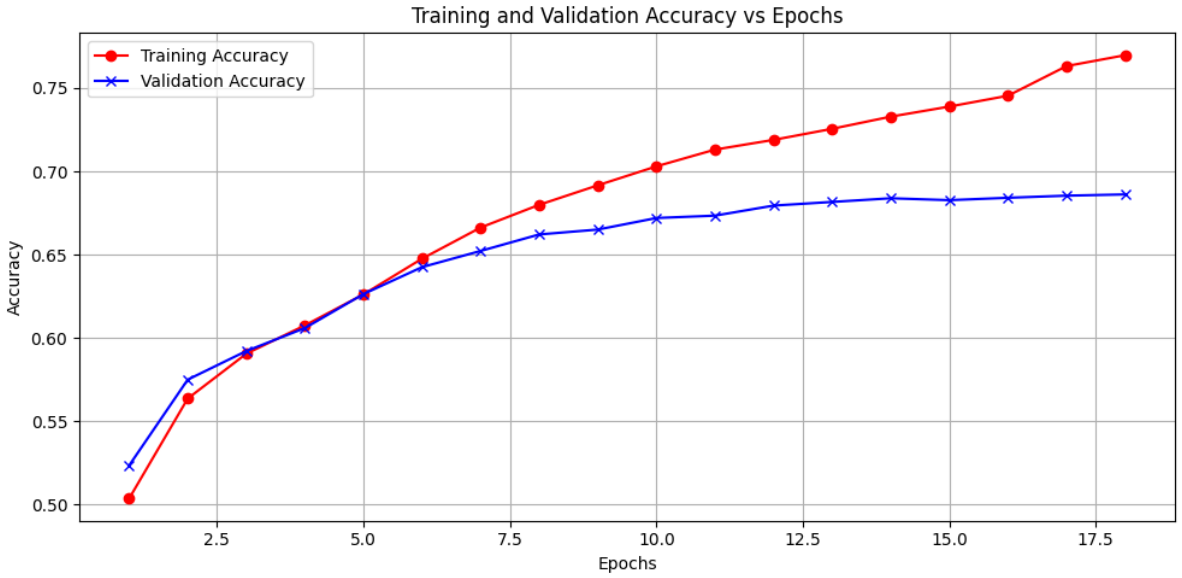


Figure 2: Training and Validation Accuracy vs Number of Epochs

Observations

- Training accuracy improved from 50.36% to 76.97% and validation accuracy from 52.34% to 68.61% over 18 epochs. Although training metrics continuously improved, validation accuracy plateaued around epochs 12-13, with validation loss increasing after epoch 13, indicating early signs of overfitting.
- This overfitting likely stems from using frozen ResNet101 features, limiting the model's ability to adapt visual representations specifically for VQA. The growing gap between training and validation metrics (particularly after epoch 13) shows that the model is specializing to training examples but struggling to generalize.

Inference

In the inference part, the trained model was evaluated on the test set. The following results were obtained:

- Loss: 0.7801
- Accuracy: 0.6838
- Precision: 0.6811
- Recall: 0.6838
- F1 Score: 0.6815

Correct Predictions

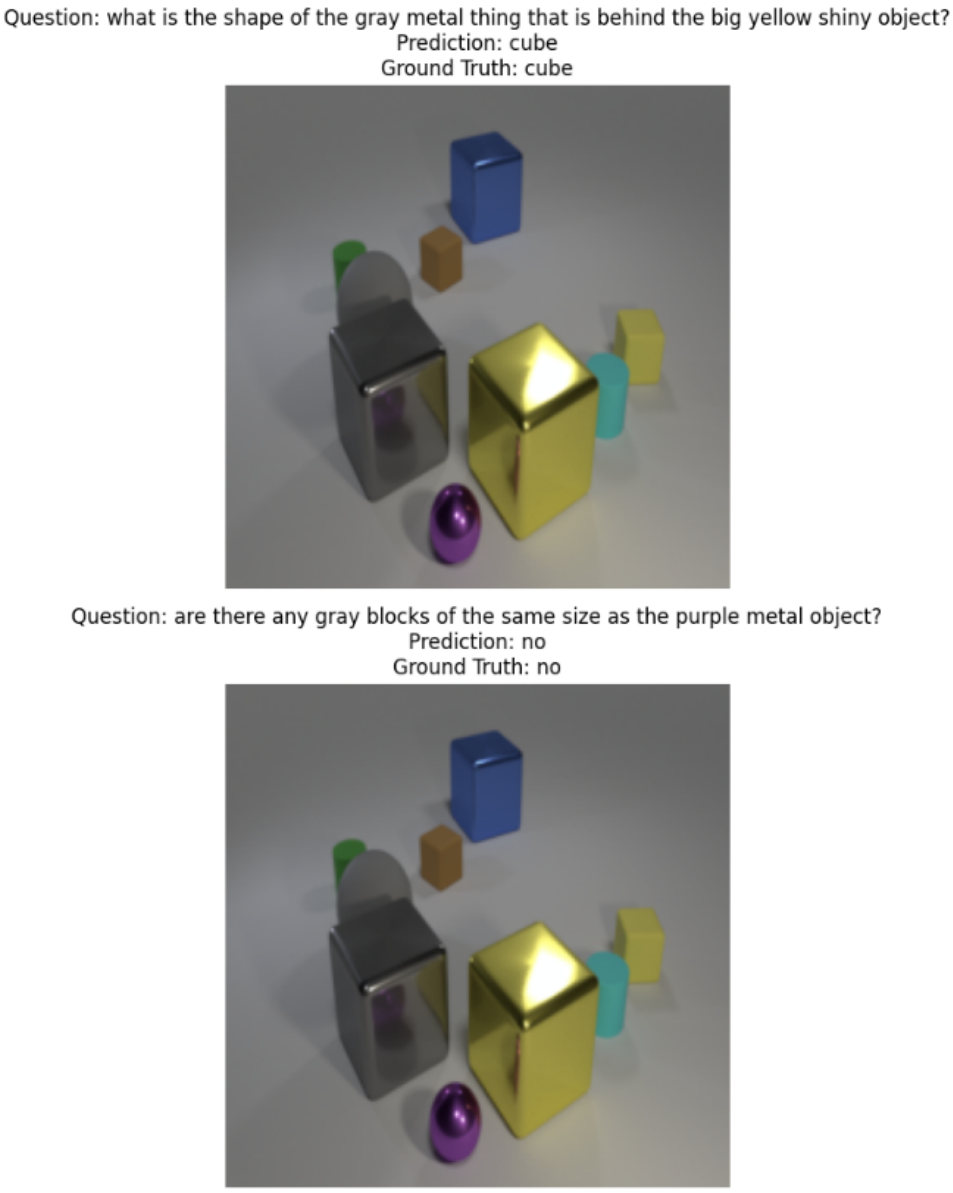
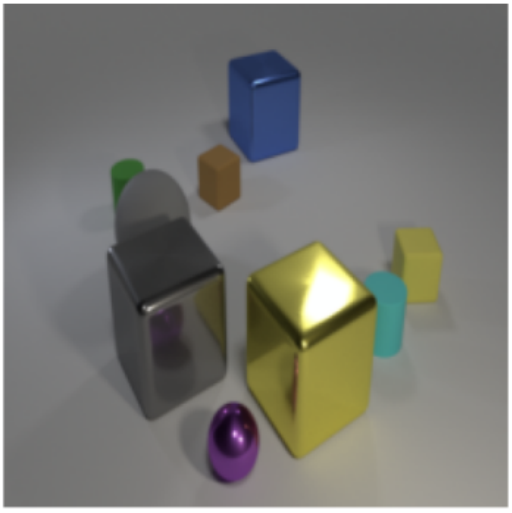
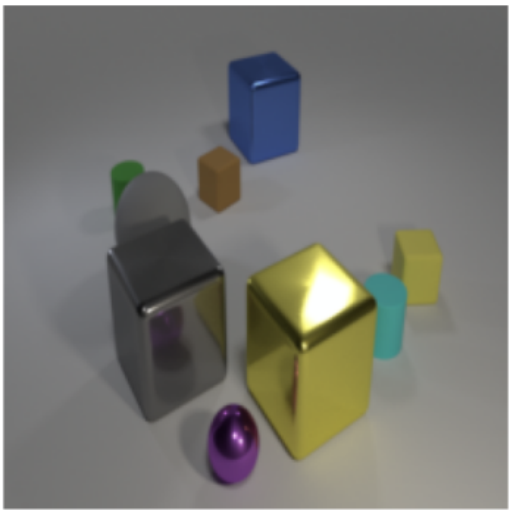


Figure 3: Correct prediction example 1

Question: is the shape of the big blue shiny thing the same as the small cyan rubber object?
Prediction: no
Ground Truth: no



Question: are there more purple shiny cylinders than matte balls?
Prediction: no
Ground Truth: no



Question: are there any large blue metal things of the same shape as the green object?
Prediction: no
Ground Truth: no

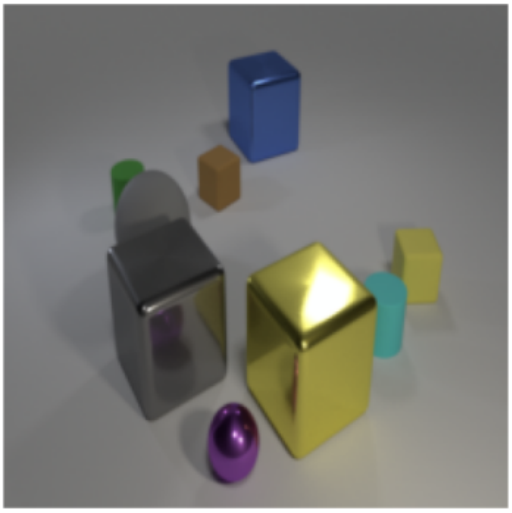
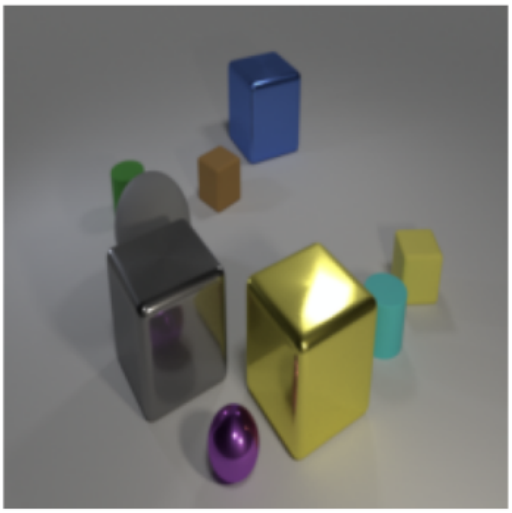


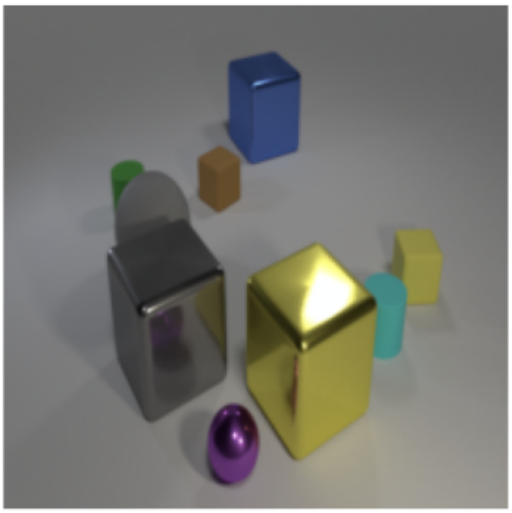
Figure 4: Correct prediction example 2

Incorrect Predictions

Question: what is the material of the small cyan thing that is the same shape as the small green rubber object?
Prediction: metal
Ground Truth: rubber



Question: there is a yellow block that is the same size as the brown rubber thing ; what material is it?
Prediction: metal
Ground Truth: rubber



Question: what number of yellow things are behind the tiny cyan cylinder?
Prediction: 0
Ground Truth: 1

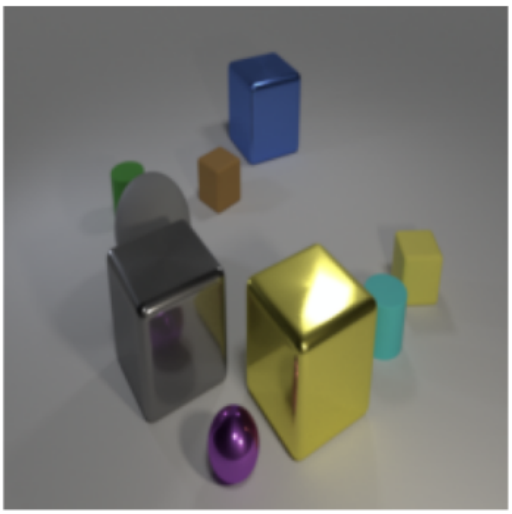
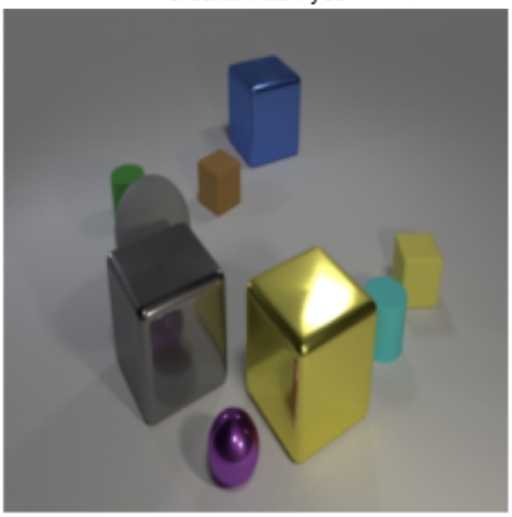


Figure 5: Incorrect prediction example 1

Question: is the size of the purple ball the same as the green matte cylinder?
Prediction: no
Ground Truth: yes



Question: is the object behind the tiny sphere made of the same material as the big red ball?
Prediction: yes
Ground Truth: no

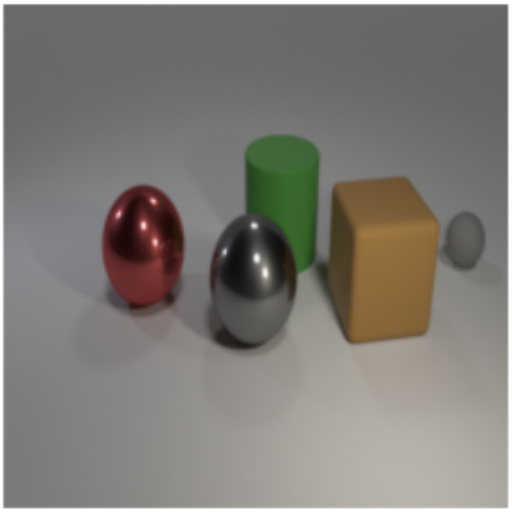


Figure 6: Incorrect prediction example 2

Part 9

- In this part, we had to use unfreeze the image encoder by setting resnet.requires grad = True.
- The training was not done from scratch, instead the best model of part 8 was used and trained further.
- Using this technique, training is completed in 5 epochs, the validation loss was increasing, so the training was stopped.

Epoch	Train Loss	Train Accuracy	Val Loss	Val Accuracy
0	0.5973	0.7328	0.7469	0.6838
1	0.6431	0.7118	0.5896	0.7340
2	0.5577	0.7465	0.5479	0.7540
3	0.5115	0.7685	0.5249	0.7629
4	0.4759	0.7870	0.5424	0.7676
5	0.4454	0.8026	0.5891	0.7640

Table 2: Train and Validation metrics - Part 9

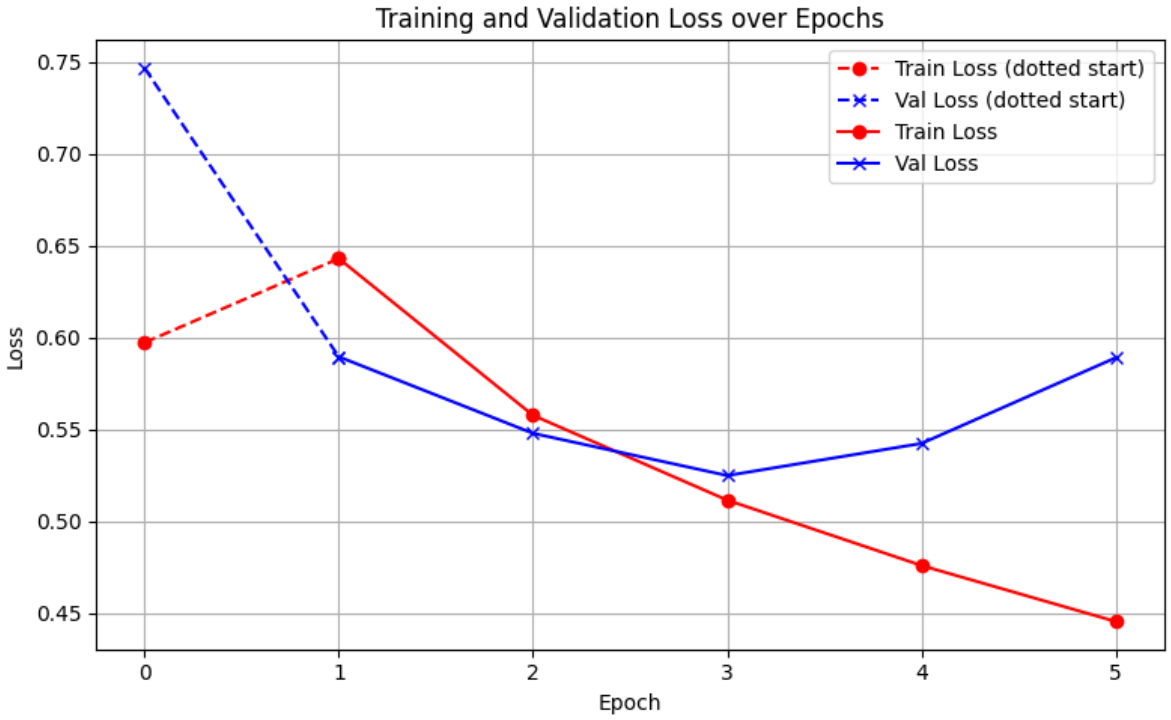


Figure 7: Training and Validation Loss vs Number of Epochs

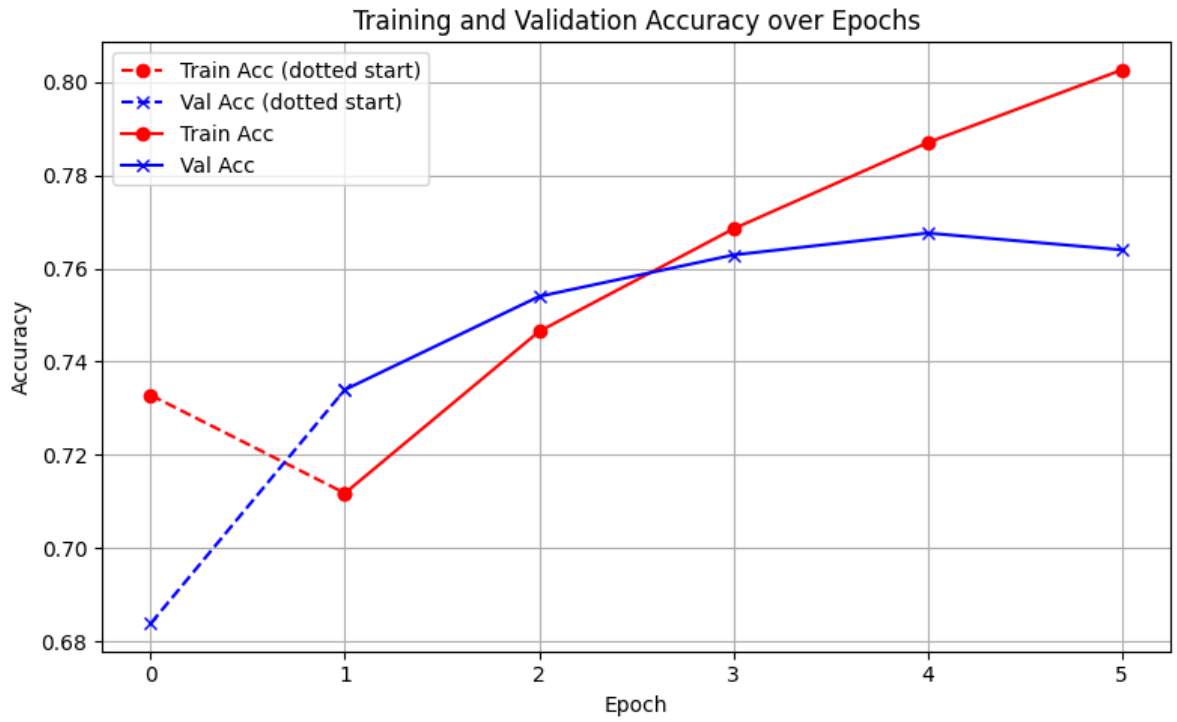


Figure 8: Training and Validation Accuracy vs Number of Epochs

Observations

- After unfreezing the ResNet101 image encoder, the validation accuracy improved significantly from 68. 38% to 76. 40% in just 5 epochs, with the test accuracy reaching 76.97%. However, after epoch 4, validation loss began to increase, while training loss continued to decrease, indicating the onset of overfitting.
- The substantial performance improvement (approximately 8% accuracy gain) demonstrates that fine-tuning the image encoder allows the model to adapt visual feature extraction specifically to the CLEVR dataset's characteristics. This confirms that the frozen ResNet101 was indeed a bottleneck in Part 8, limiting the model's ability to extract task-specific visual features necessary for optimal VQA performance.

Inference

In the inference part, the trained model was evaluated on the test set. The following results were obtained:

- Loss: 0.5457
- Accuracy: 0.7697
- Precision: 0.7643
- Recall: 0.7697
- F1 Score: 0.7656

Comments on Test Metrics

The model showed substantial improvement from Part 8 to Part 9, with the test precision increasing from 68. 38% to 76. 97%, the precision from 68. 11% to 76. 43%, the recall from 68. 38% to 76. 97%, and the F1 score from 68.15% to 76.56%. This significant performance boost of approximately 8.5% across all metrics can be attributed to the unfreezing of the ResNet101 image encoder, which allowed the model to fine-tune visual feature extraction specifically for the CLEVR dataset instead of relying on generic pre-trained features. The ability to adapt the entire visual processing pipeline to the specific characteristics of the CLEVR images enabled the model to extract more relevant and discriminative visual features, greatly improving its ability to answer questions accurately.

Correct Predictions

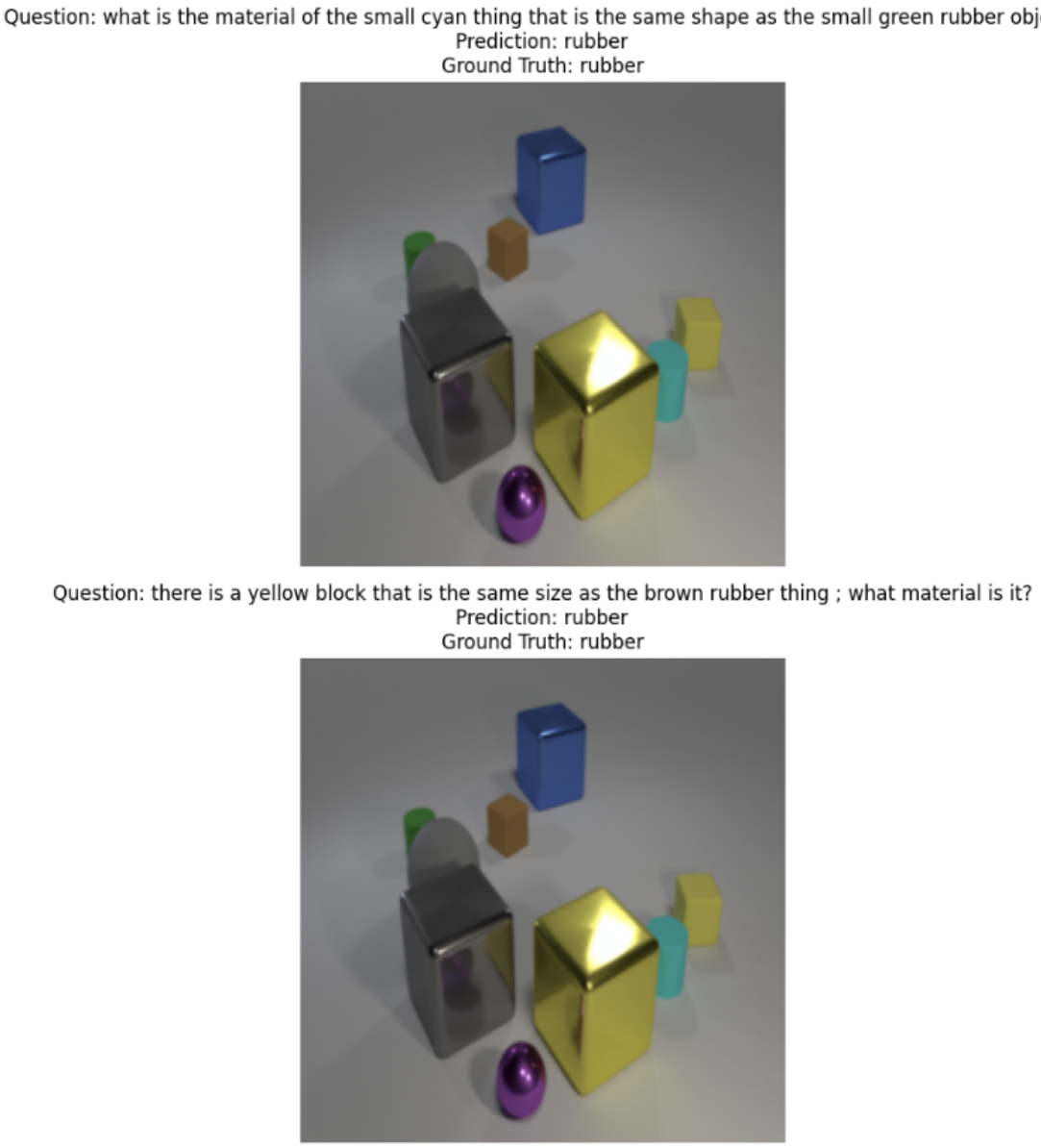
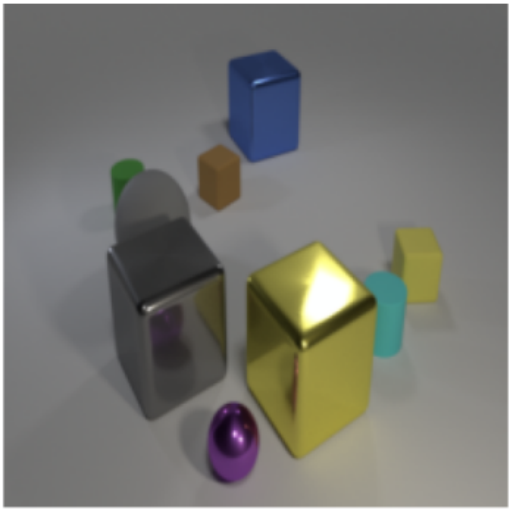
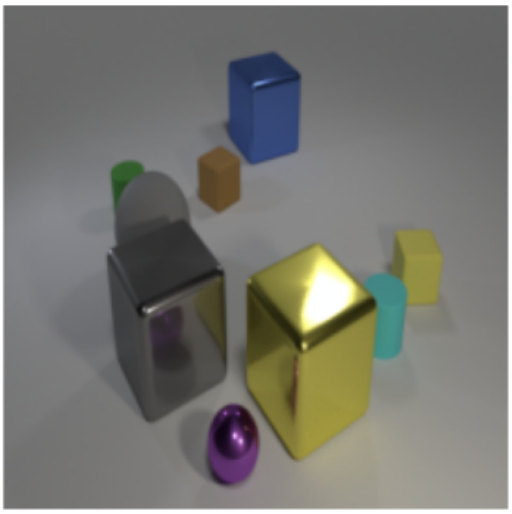


Figure 9: Correct prediction example 1

Question: is the shape of the big blue shiny thing the same as the small cyan rubber object?
Prediction: no
Ground Truth: no



Question: are there more purple shiny cylinders than matte balls?
Prediction: no
Ground Truth: no



Question: are there any large blue metal things of the same shape as the green object?
Prediction: no
Ground Truth: no

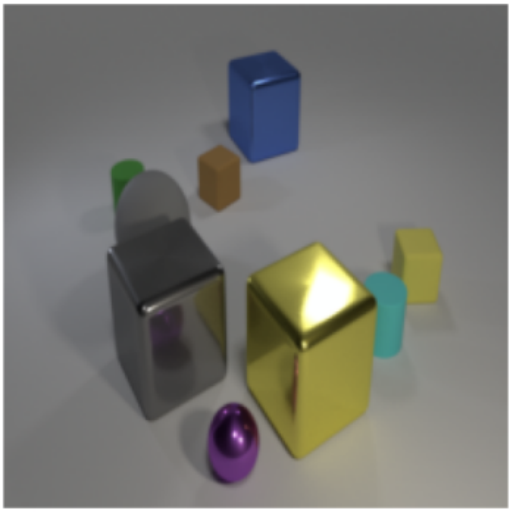
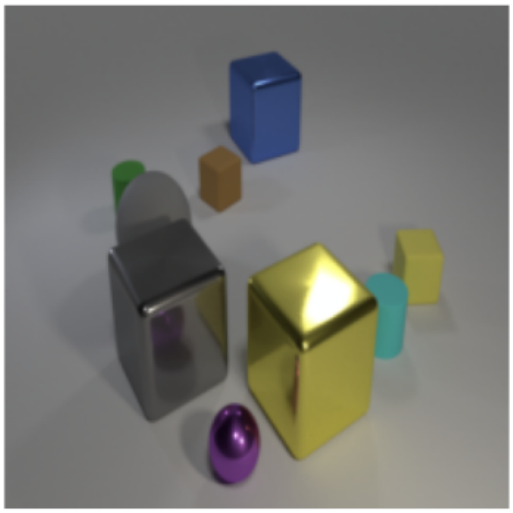


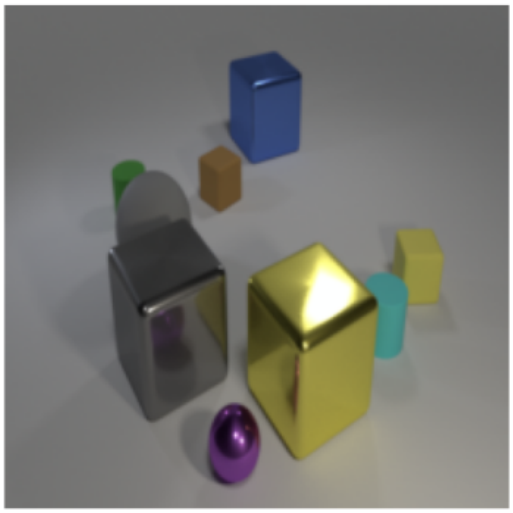
Figure 10: Correct prediction example 2

Incorrect Predictions

Question: what is the shape of the gray metal thing that is behind the big yellow shiny object?
Prediction: sphere
Ground Truth: cube



Question: are there any gray blocks of the same size as the purple metal object?
Prediction: yes
Ground Truth: no



Question: what number of yellow things are behind the tiny cyan cylinder?
Prediction: 0
Ground Truth: 1

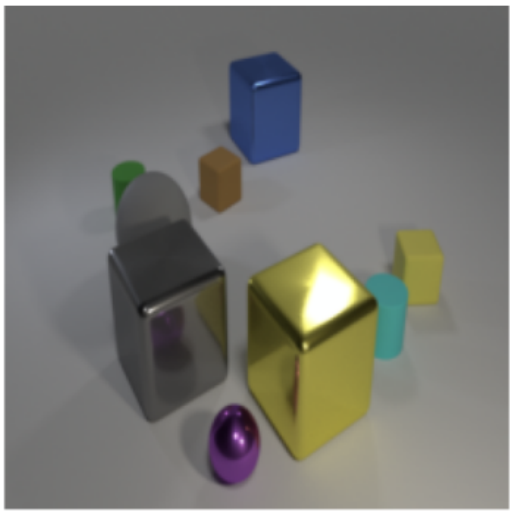
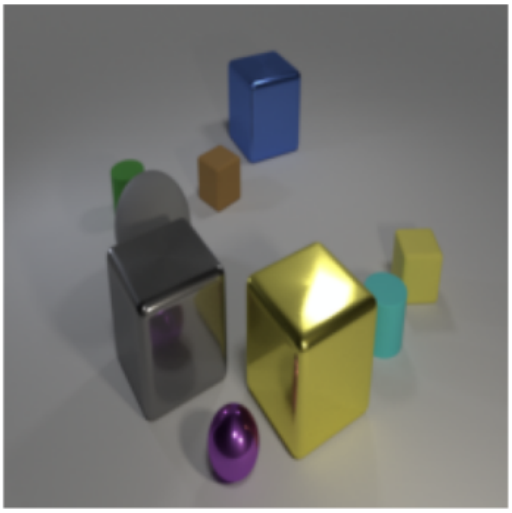


Figure 11: Incorrect prediction example 1

Question: what is the material of the thing that is the same color as the rubber sphere?
Prediction: rubber
Ground Truth: metal



Question: what color is the other big thing that is the same shape as the gray metal object?
Prediction: brown
Ground Truth: red

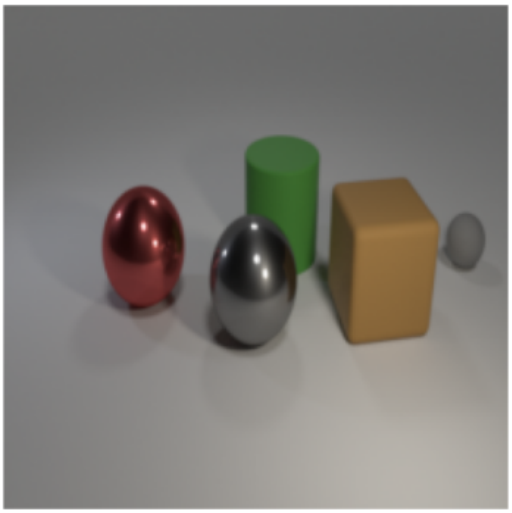


Figure 12: Incorrect prediction example 2

Part 10a

- In this part, we unfreeze the image encoder by setting `resnet.requires_grad = True`.
- The training was not done from scratch; instead, the best model from Part 9 was used and further fine-tuned.
- Further training was performed using the Focal Loss. Focal Loss is mainly used to address class imbalance. It modifies the standard cross-entropy loss by introducing a modulating factor $(1 - p_t)^\gamma$, where p_t is the predicted probability of the correct class, and γ is a focusing parameter.
- Using this technique, training completed in 5 epochs. As the validation loss was increasing, early stopping was triggered.

Epoch	Train Loss	Train Acc	Val Loss	Val Acc
0	0.4454	0.8026	0.5891	0.7619
1	0.1510	0.8281	0.2174	0.7903
2	0.1242	0.8519	0.2228	0.7905
3	0.1046	0.8707	0.2440	0.7916
4	0.0895	0.8856	0.2855	0.7878
5	0.0631	0.9172	0.3494	0.7904

Table 3: Train and Validation metrics - Part 10a

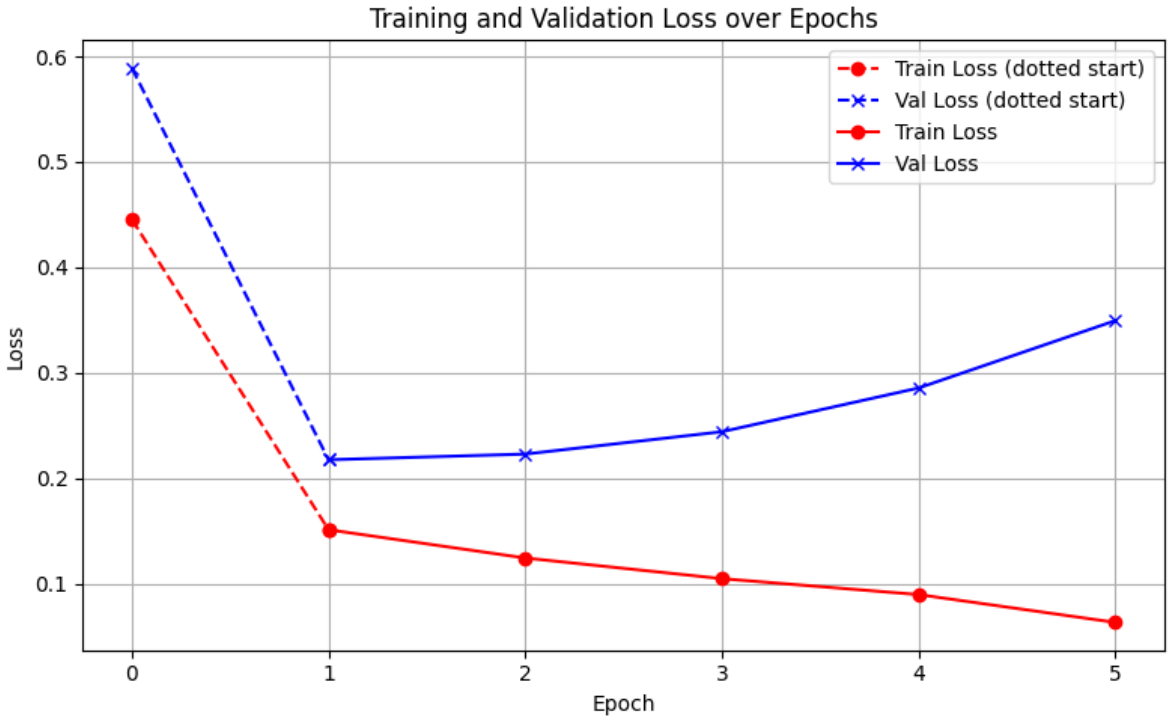


Figure 13: Training and Validation Loss vs Number of Epochs

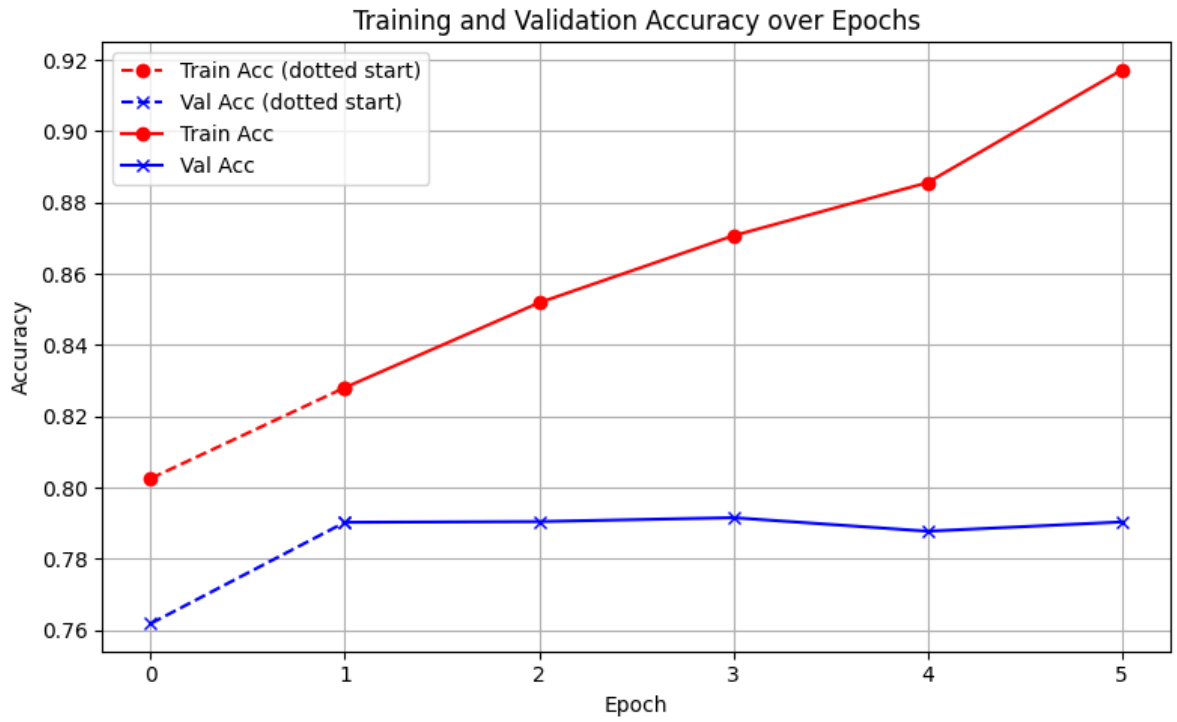


Figure 14: Training and Validation Accuracy vs Number of Epochs

Observations

- Implementing Focal Loss led to significant improvements with validation accuracy increasing from 76.19% to 79.04% and test accuracy reaching 79.32%. However, a substantial gap emerged between training and validation loss curves, with training loss decreasing to 0.0631 while validation loss increased to 0.3494 by epoch 5.
- The performance boost can be attributed to Focal Loss's ability to address class imbalance by focusing more on difficult examples while downweighting well-classified examples. This allows the model to better handle the inherent imbalance in answer distributions in the CLEVR dataset, but the widening gap between training and validation metrics indicates that the model is becoming increasingly specialized to the training data.

Inference

In the inference part, the trained model was evaluated on the test set. The following results were obtained:

- Loss: 0.2476
- Accuracy: 0.7932
- Precision: 0.7924
- Recall: 0.7932
- F1 Score: 0.7925

Comment on Test Metrics

From Part 9 to Part 10a, the model demonstrated further improvement with test precision increasing from 76. 97% to 79. 32%, precision from 76. 43% to 79. 24%, recall from 76. 97% to 79. 32%, and F1 score from 76.56% to 79.25%. This additional improvement of approximately 2. 3% in all metrics resulted from the implementation of focal loss, which specifically addresses class imbalance by focusing training on difficult examples while downweighting well-classified samples. The CLEVR dataset likely contains imbalanced answer distributions, and Focal Loss helped the model better handle these imbalances by preventing easy, common examples from dominating the training process, thus improving its ability to correctly classify all answer categories.

Correct Predictions

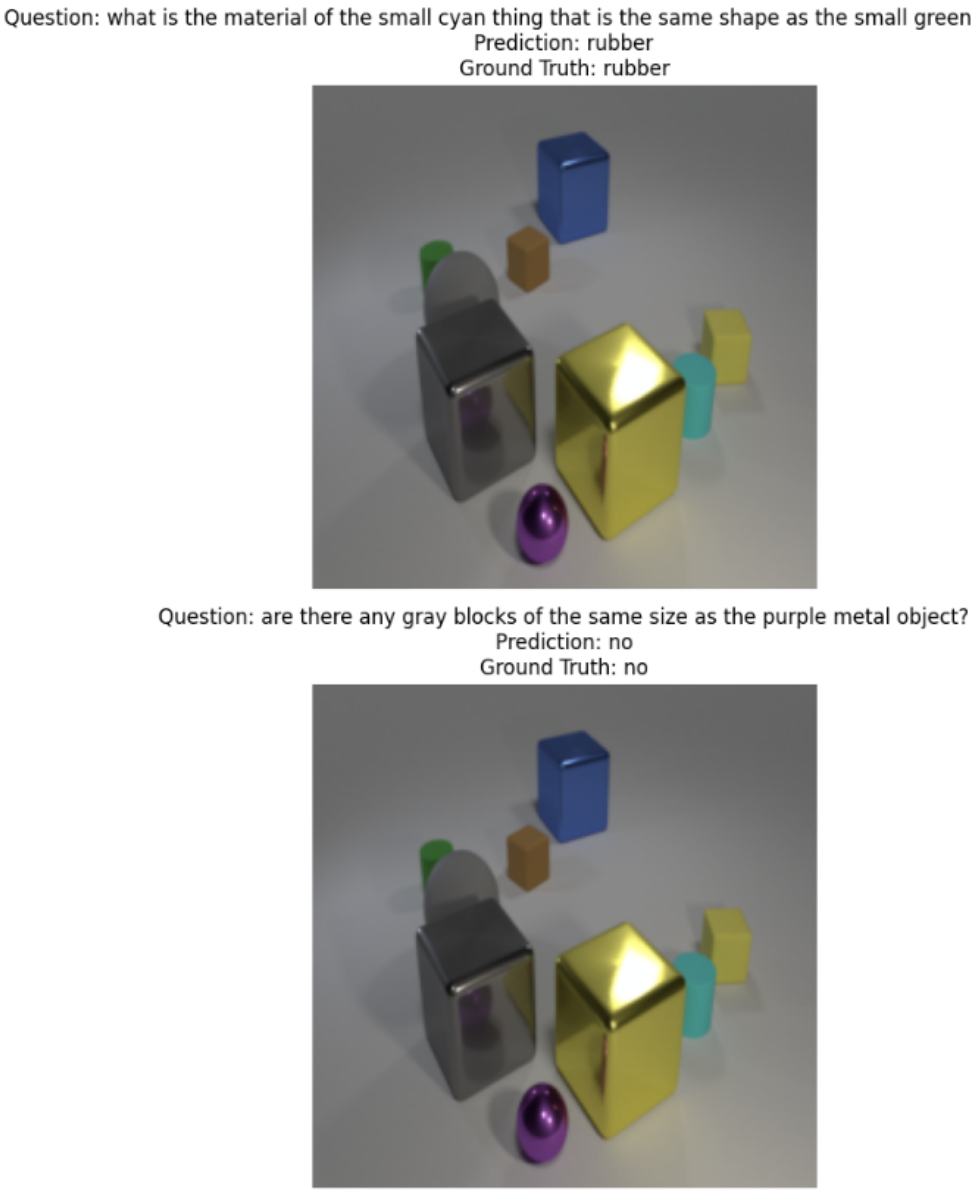
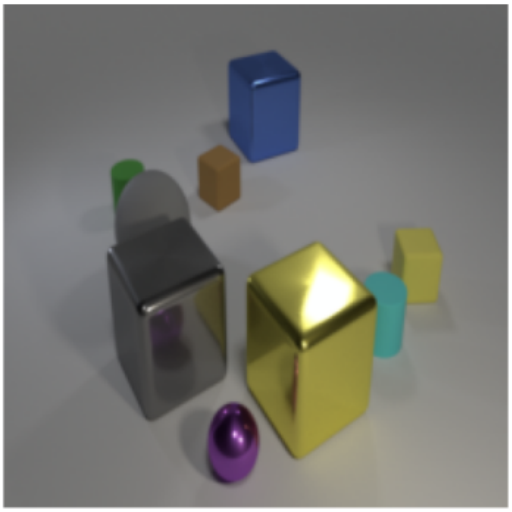
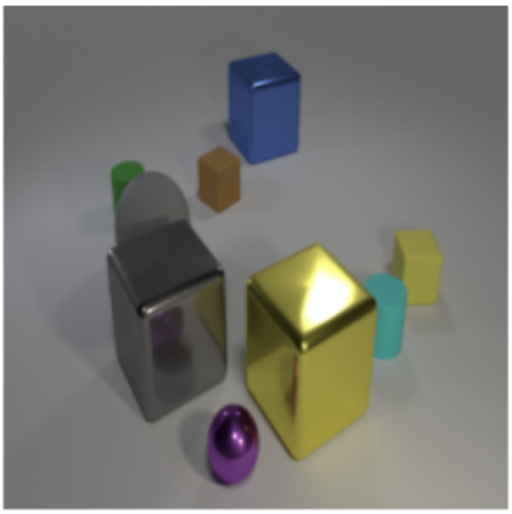


Figure 15: Correct prediction example 1

Question: there is a yellow block that is the same size as the brown rubber thing ; what material is it?
Prediction: rubber
Ground Truth: rubber



Question: is the shape of the big blue shiny thing the same as the small cyan rubber object?
Prediction: no
Ground Truth: no



Question: are there more purple shiny cylinders than matte balls?
Prediction: no
Ground Truth: no

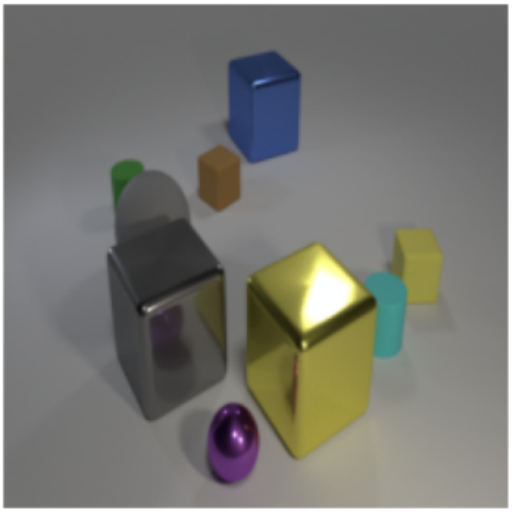
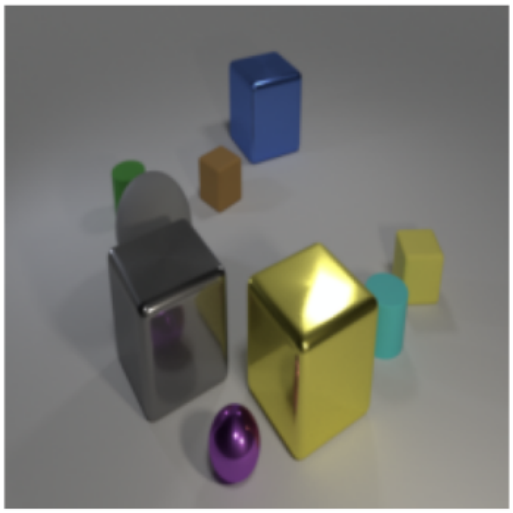


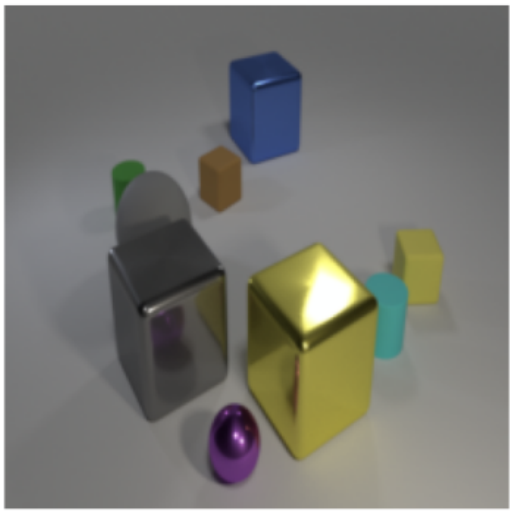
Figure 16: Correct prediction example 2

Incorrect Predictions

Question: what is the shape of the gray metal thing that is behind the big yellow shiny object?
Prediction: sphere
Ground Truth: cube



Question: what number of yellow things are behind the tiny cyan cylinder?
Prediction: 0
Ground Truth: 1



Question: what is the material of the thing that is the same color as the rubber sphere?
Prediction: rubber
Ground Truth: metal

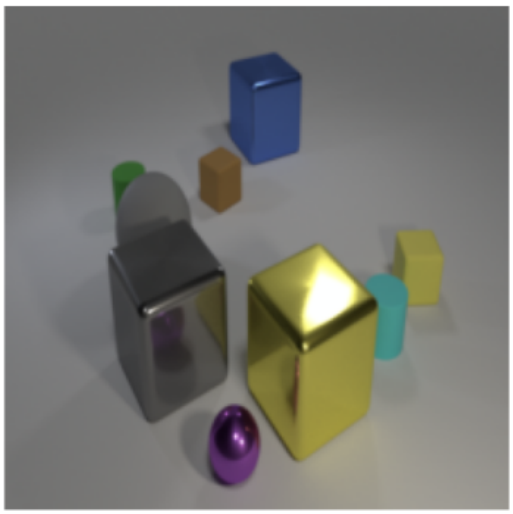
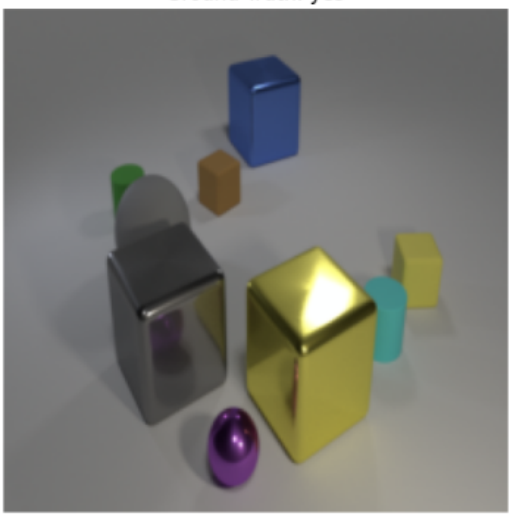


Figure 17: Incorrect prediction example 1

Question: is the size of the purple ball the same as the green matte cylinder?
Prediction: no
Ground Truth: yes



Question: what color is the other big thing that is the same shape as the gray metal object?
Prediction: brown
Ground Truth: red

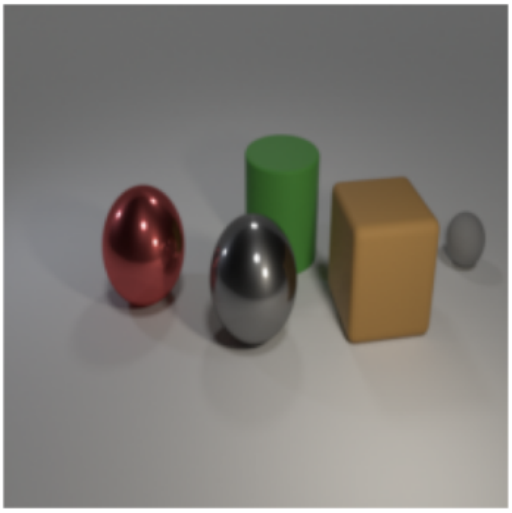


Figure 18: Incorrect prediction example 2

Part 10b

- The conditions are similar to 10a. Now, proceed by initializing the word embedding layer (nn.Embedding) with BERT embeddings instead of random initialization.
- The training was not done from scratch; instead, the best model from Part 10a was used and further fine-tuned.
- Training is taking a lot of epochs because fine-tuning large models like BERT requires significant computation, especially if all layers are being updated, leading to slower convergence. Hence, due to time constraints, we were able to train only upto 8 epochs, and we can see that the validation accuracy is increasing at a steady rate

Epoch	Train Loss	Train Acc	Val Loss	Val Acc
1	0.8012	0.6123	1.1042	0.5253
2	0.7250	0.6554	1.0156	0.5629
3	0.6827	0.6765	0.9453	0.5937
4	0.6484	0.6947	0.8767	0.6176
5	0.5612	0.7407	0.8456	0.6392
6	0.4362	0.8039	0.8234	0.6442
7	0.4009	0.8208	0.8090	0.6534
8	0.3778	0.8321	0.7880	0.6690

Table 4: Train and Validation metrics - Part 10b

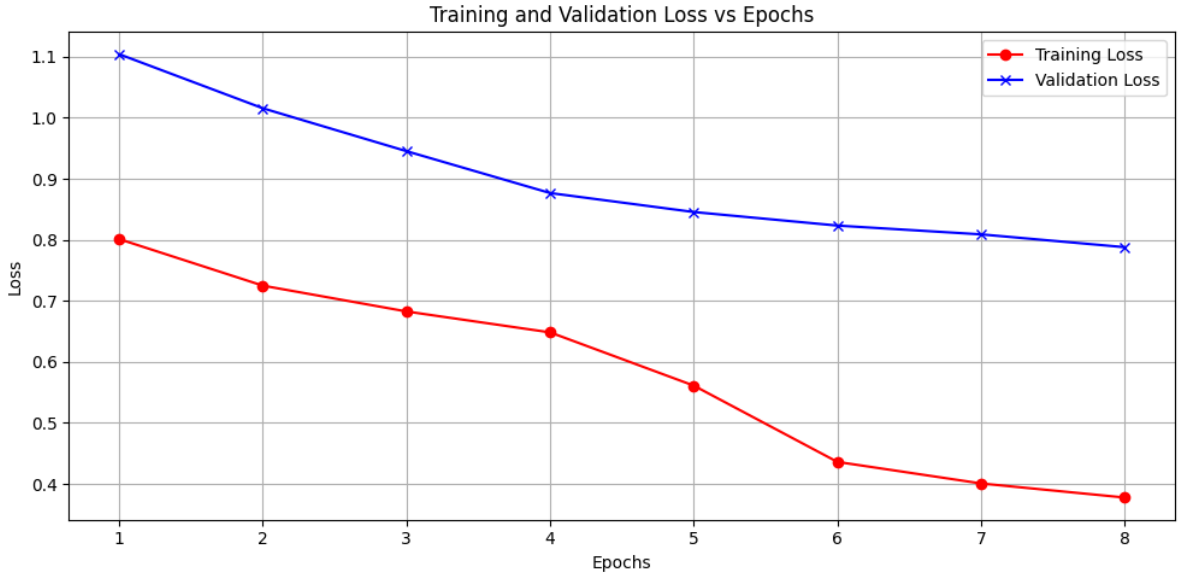


Figure 19: Training and Validation Loss vs Number of Epochs

Epoch Train Loss Train Acc Val Loss Val Acc

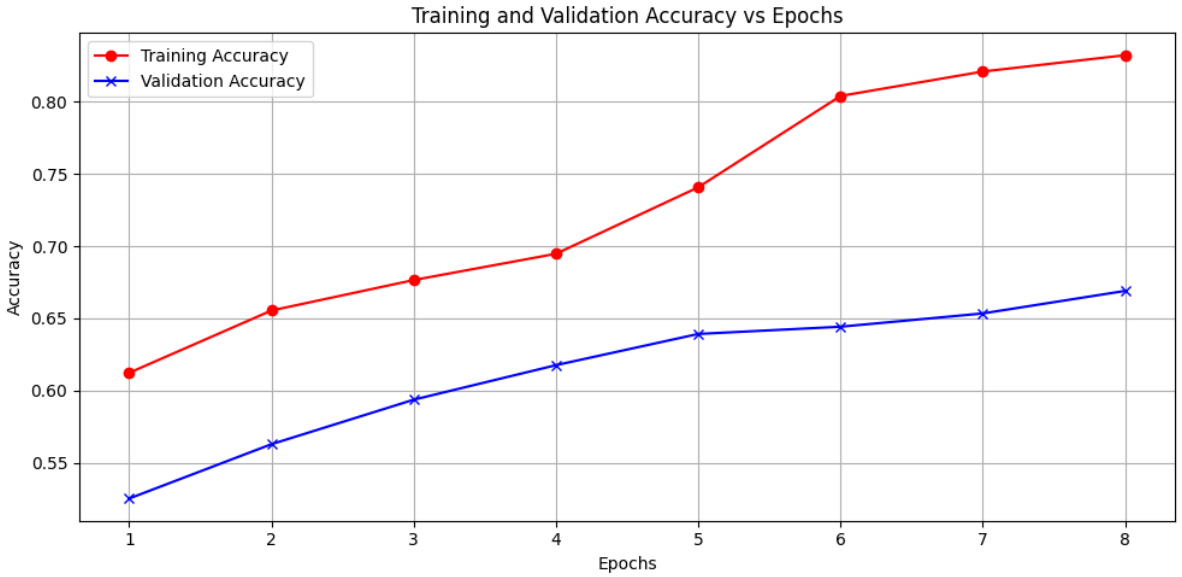


Figure 20: Training and Validation Accuracy vs Number of Epochs

Observations

- Despite starting from a high-performance model (79.32% test accuracy in Part 10a), initializing with BERT embeddings caused a drop in performance, though both metrics steadily improved over 8 epochs, reaching 66.90% validation accuracy with a clear upward trajectory suggesting continued improvement with additional training.
- The performance drop occurs because BERT embeddings were pre-trained on general text different from CLEVR questions, requiring the model to relearn relationships between word representations and visual features. The steady improvement shows the model is adapting these richer linguistic representations, with potential to exceed part-10a's performance given sufficient training time.

Inference

In the inference part, the trained model was evaluated on the test set. The following results were obtained (NOTE: the model was only trained until epoch = 8):

- Loss: 0.9912
- Accuracy: 0.6476
- Precision: 0.6445
- Recall: 0.6476
- F1 Score: 0.6447

Comment on Test Metrics

The transition from Part 10a to Part 10b currently shows a performance decrease with test accuracy dropping from 79.32% to 64.76%, but this is expected as the model was only trained for 8 epochs after a significant architectural change. Despite this temporary decline, the steady upward trajectory in both training and validation accuracy (from 51.53% to 66.90% validation accuracy over just 8 epochs) suggests strong potential for improvement with additional training. The integration of BERT embeddings provides richer linguistic representations than random initialization, potentially enabling more nuanced understanding of question semantics. The model is likely to surpass the performance of 10a by leveraging BERT's powerful contextual understanding of language. (given a higher training time)

Correct Predictions

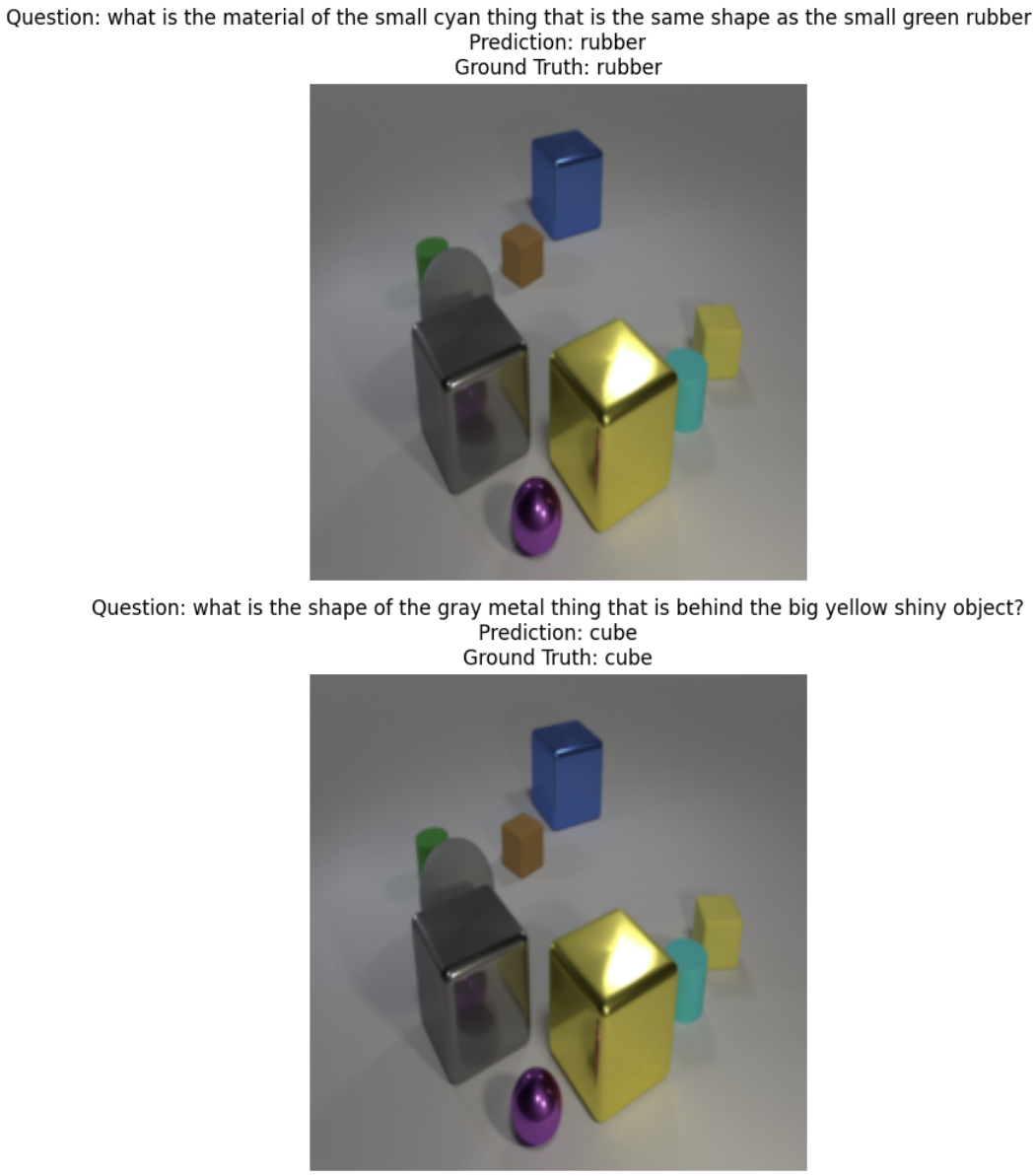
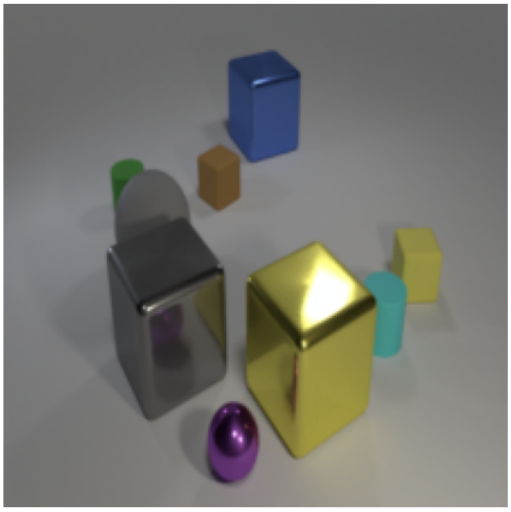
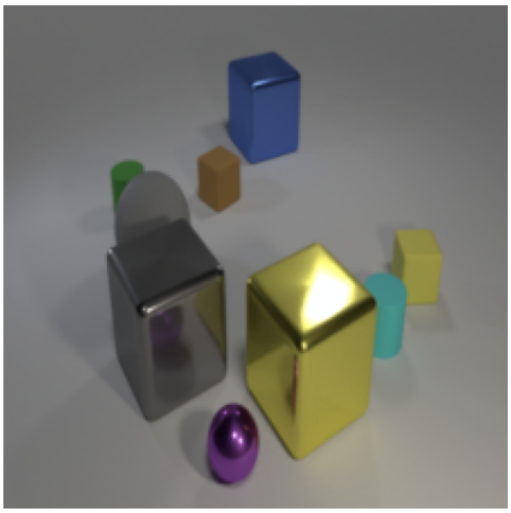


Figure 21: Correct prediction example 1

Question: is the shape of the big blue shiny thing the same as the small cyan rubber object?
Prediction: no
Ground Truth: no



Question: are there more purple shiny cylinders than matte balls?
Prediction: no
Ground Truth: no



Question: are there any large blue metal things of the same shape as the green object?
Prediction: no
Ground Truth: no

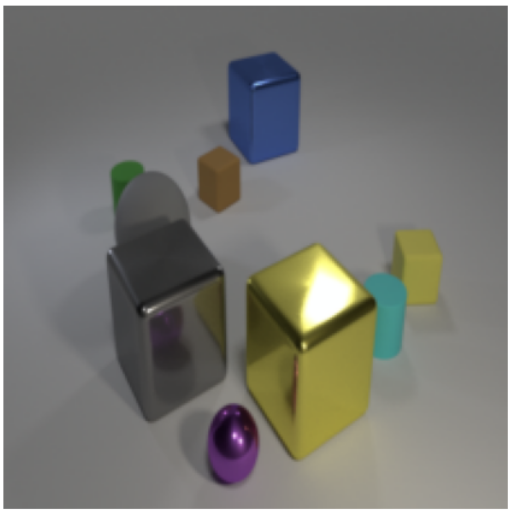
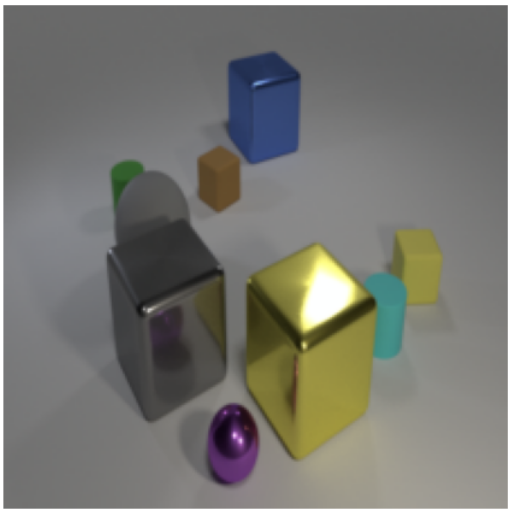


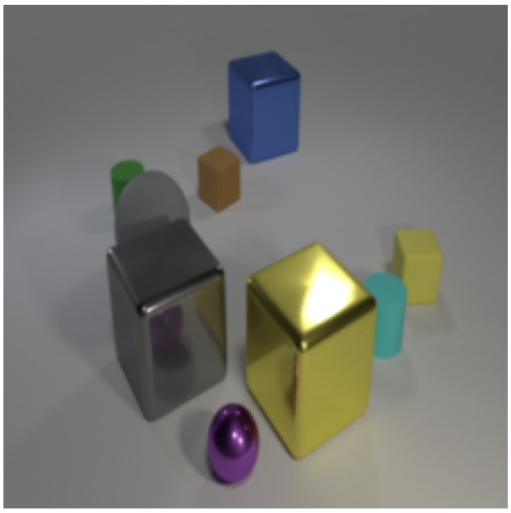
Figure 22: Correct prediction example 2

Incorrect Predictions

Question: are there any gray blocks of the same size as the purple metal object?
Prediction: yes
Ground Truth: no



Question: there is a yellow block that is the same size as the brown rubber thing ; what material is it?
Prediction: metal
Ground Truth: rubber



Question: what number of yellow things are behind the tiny cyan cylinder?
Prediction: 0
Ground Truth: 1

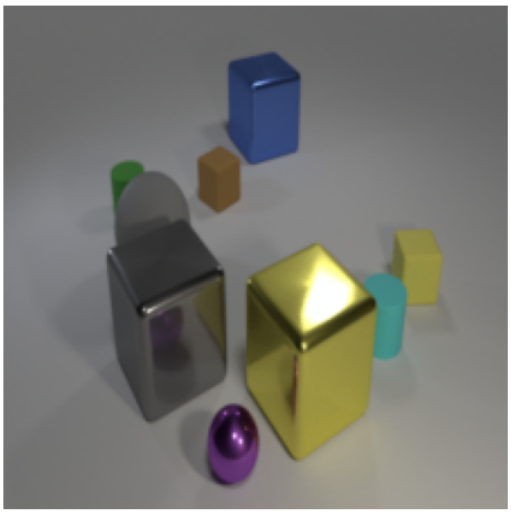
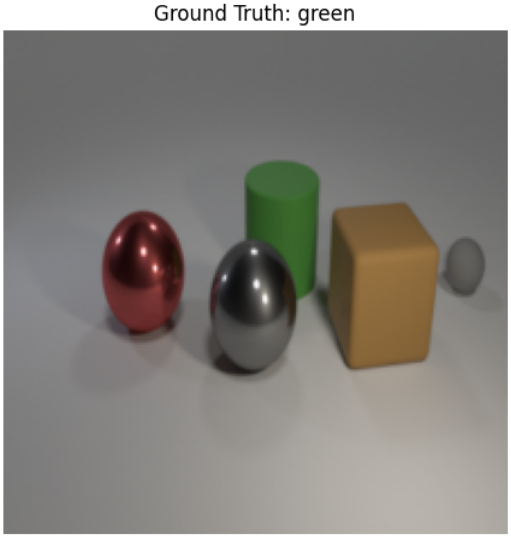


Figure 23: Incorrect prediction example 1

:tion: what color is the big rubber object that is behind the gray ball that is right of the big object behind the gray matte
Prediction: brown
Ground Truth: green



Question: what color is the other big thing that is the same shape as the gray metal object?
Prediction: gray
Ground Truth: red

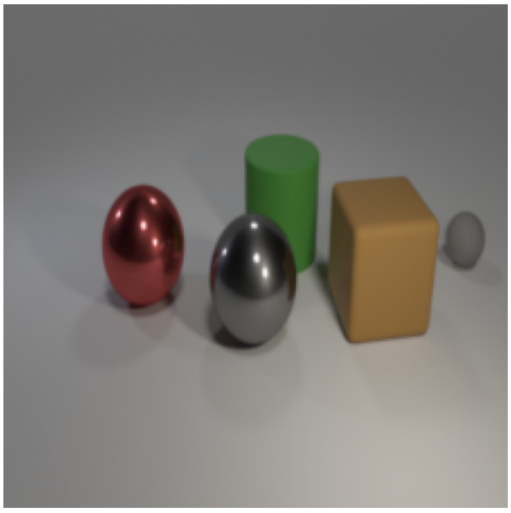


Figure 24: Incorrect prediction example 2

Part 11

- In this part, we evaluate the model's ability to generalize across variations in object color and shape.
- To do this, we evaluated the best model trained on the type A dataset on the type B dataset without any additional training.
- The results obtained are shown.

Inference

In the inference part, the trained model for dataset A was evaluated on the test set B. The following results were obtained:

- Accuracy: 0.6705
- Precision: 0.6626
- Recall: 0.6705
- F1 Score: 0.6634

Metric	**Test Set A**	**Test Set B**
Accuracy	0.7932	0.6705
Precision	0.7924	0.6626
Recall	0.7932	0.6705
F1 Score	0.7925	0.6634

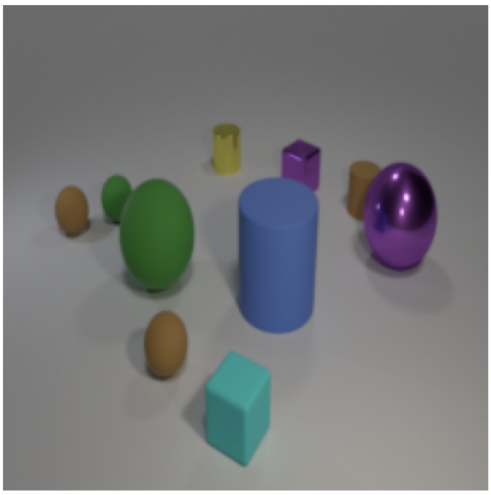
Table 5: Comparison of evaluation metrics between Test Set A and Test Set B

Conclusion

The zero-shot evaluation on Test Set B demonstrates a noticeable drop in performance compared to Test Set A, across all metrics — accuracy, precision, recall, and F1 score. This decline indicates that although the model performs well on the distribution it was trained on (type A), it **struggles to generalize** to the unseen variations present in the type B dataset, particularly differences in object colors and shapes.

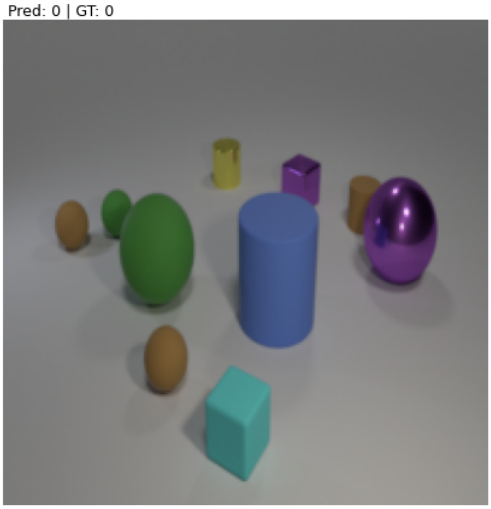
This outcome emphasizes the importance of *diverse and representative training data* in visual question answering tasks. It also suggests that incorporating domain adaptation techniques or training on a more varied dataset could improve generalization capabilities. Overall, the results underline a key limitation of the current model in zero-shot settings, providing a clear direction for future enhancements.

Correct Predictions



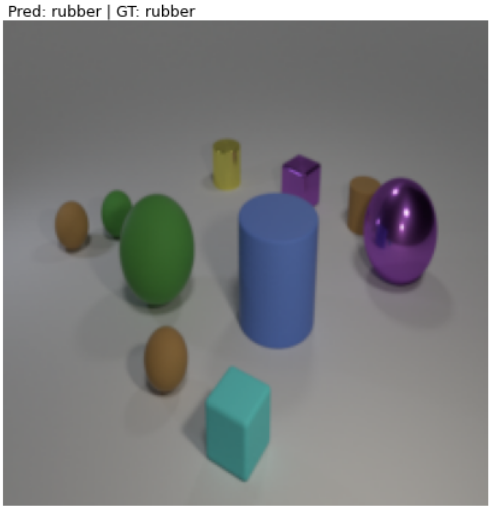
Q: what number of small metallic things are on the right side of the large sphere that is on the right side of the big rubber thing on the

Pred: 0 | GT: 0



Q: what material is the cyan thing?

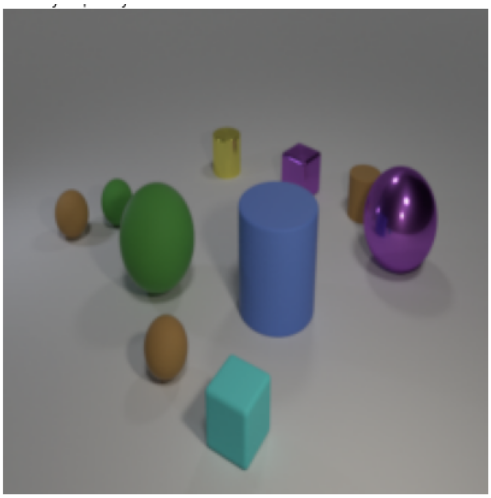
Pred: rubber | GT: rubber



Q: there is a brown ball behind the blue matte cylinder ; is it the same size as the yellow cylinder?

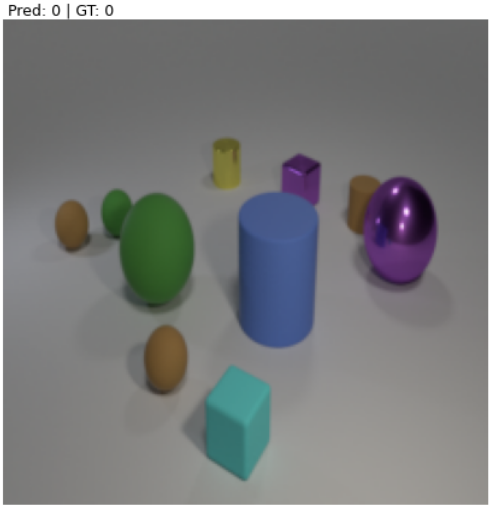
Pred: yes | GT: yes

Figure 25: Correct prediction example 1



Q: what number of other rubber cubes are the same color as the matte cube?

Pred: 0 | GT: 0

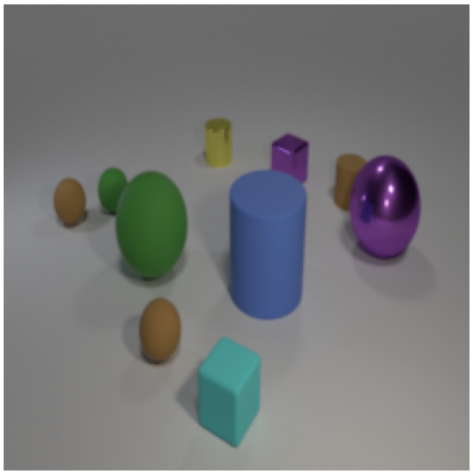


Q: are there more tiny brown matte things that are on the right side of the metallic block than blue rubber things?

Pred: no | GT: no

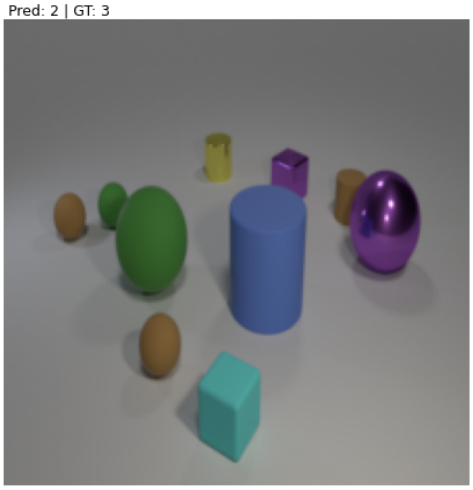
Figure 26: Correct prediction example 2

Incorrect Predictions



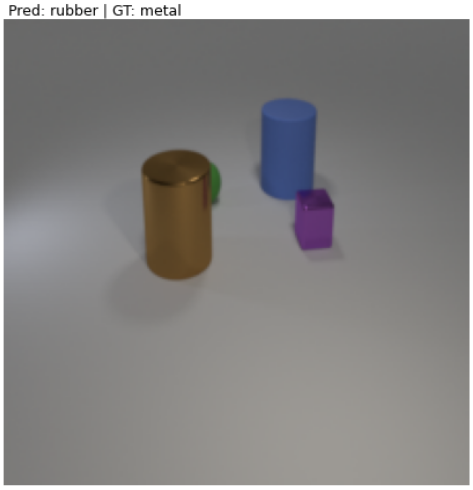
Q: what number of things are either balls that are left of the big green rubber ball or big blue matte things?

Pred: 2 | GT: 3



Q: what material is the tiny cube that is the same color as the big metal object?

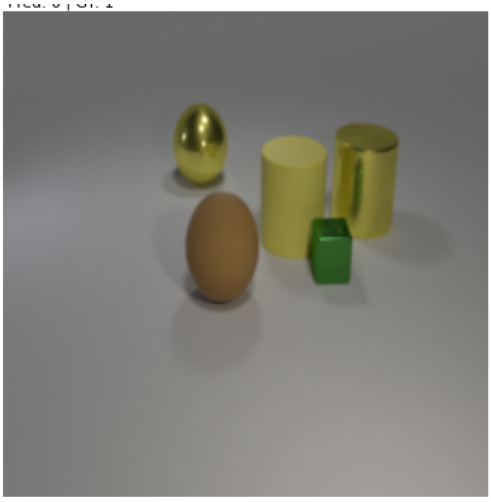
Pred: rubber | GT: metal



Q: how many other things are the same size as the green rubber object?

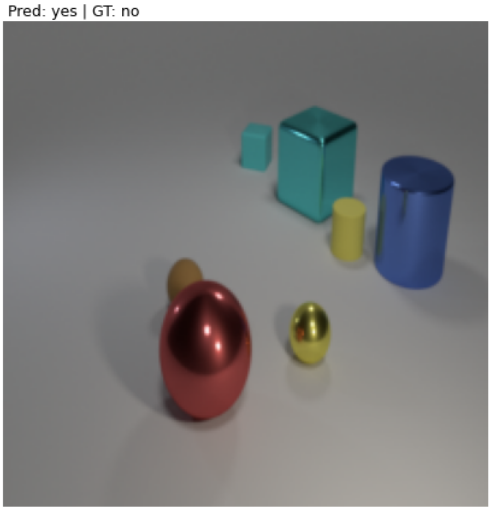
Pred: 0 | GT: 1

Figure 27: Incorrect prediction example 1



Q: is there anything else that has the same shape as the small metal thing?

Pred: yes | GT: no



Q: there is a ball that is both left of the yellow metal object and in front of the tiny rubber sphere ; how big is it?

Pred: small | GT: large

Figure 28: Incorrect prediction example 2

Final Model

Link to the drive where models are uploaded : `https://drive.google.com/drive/folders/1SDF0-IVOuaR1BNzDzon4gfV8BBzfhixE?usp=sharing`