Enhancing Visual Question Answering (VQA) with Deep Reasoning Networks (DRNs)

Gonzalo de Hermenegildo
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
gdeherme@andrew.cmu.edu

Abstract

This paper presents an enhanced approach to the Visual Question Answering (VQA) task by integrating Deep Reasoning Networks (DRNs) to augment the reasoning capabilities of deep learning models. While traditional deep learning architectures excel in pattern recognition, they often lack the structured reasoning required for complex tasks such as VQA. Our proposed method extends standard deep learning models with two additional modules: a Neural Logical Reasoning Module, which combines symbolic logic with neural processing to support multi-step deductions, and a Graph Neural Network Module that represents spatial relationships between image objects, enabling a deeper understanding of object interactions and geometric configurations. We evaluate our models on a modified version of the MSCOCO dataset, assessing metrics such as test accuracy, amount of data needed to learn, learning speed, and robustness to noise. The experimental results demonstrate that our models outperform the baseline approach, particularly in scenarios with limited training data and noisy inputs.

1 Introduction

Human reasoning operates via two distinct systems, as described by the dual-process theory: System 1, which is fast, intuitive, and automatic, and System 2, which is slow, analytical, and deliberate [1]. While System 1 facilitates rapid decisions in routine situations, System 2 is essential for deep reasoning and problem-solving in complex tasks. Most deep learning models predominantly emulate System 1, excelling in pattern recognition tasks such as image classification and speech recognition, but they often lack the structured reasoning capabilities characteristic of System 2.

Visual Question Answering (VQA) is a challenging task that requires models to comprehend and reason about visual content to answer questions accurately. Traditional deep learning models struggle with VQA because it demands both pattern recognition and structured reasoning. Enhancing deep learning models with reasoning capabilities akin to System 2 could significantly improve their performance on such tasks.

In this paper, we propose an approach to enhance VQA performance by integrating Deep Reasoning Networks (DRNs) with standard deep learning architectures. DRNs are designed to combine the pattern recognition strengths of deep learning (System 1) with reasoning modules that enable logical deductions and multi-step reasoning (System 2).

We evaluate our approach using a modified version of the MSCOCO Multiple Choice Question and Answer dataset [2], specifically designed to test a model's reasoning capabilities through compositional language and visual reasoning tasks. The dataset consists of 4,000 images depicting simple

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

geometric shapes (e.g., triangles, circles, rectangles) in various colors, along with a set of deductive questions and answers concerning the positioning and colors of these shapes.

Our objectives are as follows:

- Develop a VQA model(s) that leverages DRNs to improve reasoning over visual and textual inputs, thereby outperforming baseline deep learning approaches.
- Assess the performance of the model(s) based on its ability to select the correct answer from multiple-choice options, given an input question and image.
- Compare our proposed models with a baseline model based on: ability to learn with scarce amounts of training data, robustness to noise in the data, and speed of learning.

2 Background

Our baseline model mimics traditional deep learning models, which typically consist of an image encoder and a question encoder, whose outputs are fused to predict an answer.

Baseline Model Architecture:

- **Dataset Pre-processing**: All input images are normalized using the ImageNet mean and standard deviation. Questions are tokenized, and multiple-choice answers are embedded for processing.
- **Image Encoder**: We use ResNet-18, a Convolutional Neural Network (CNN) pre-trained on ImageNet, to extract visual features. The extracted features are passed through a linear layer followed by a ReLU activation function.
- **Question Encoder**: The tokenized questions are passed through a pre-trained PyTorch LSTM layer to obtain a question representation. This representation is processed through a linear layer with ReLU activation to produce a feature vector.
- **Fusion Mechanism**: The image and question embeddings are concatenated and passed through a fully connected layer to predict the answer.
- **Loss Function**: Cross-entropy loss over the predefined answer classes is used for training.

The baseline model achieves a test accuracy of 88% on the VQA task with a large, noise-free training dataset. While effective at learning basic associations between image features and question embeddings, it lacks advanced reasoning capabilities which will be put to the test.

3 Related Work

Combining deep learning with reasoning has been a focal point in addressing complex problems that require both intuition and logic. Chen et al.[3] introduced Deep Reasoning Networks (DRNs), which integrate reasoning modules within deep learning architectures to handle logical tasks more effectively. Anthony et al.[4] explored structured decision-making with tree search methods, although these approaches often add significant computational demands.

Graph Neural Networks (GNNs) have been employed to model relationships between entities, proving useful in tasks requiring an understanding of interactions [5]. Applying GNNs to VQA allows the model to capture spatial and relational information between objects in an image.

Our work builds upon these foundational methods by integrating both logical reasoning and graph-based relational reasoning into the VQA framework, aiming to enhance performance on tasks requiring multi-step reasoning and improved interpretability.

4 Methods

We believe integrating logical and deep reasoning techniques will enhance the baseline model's ability to comprehend and infer complex geometric relationships and object properties. Which will make it more effective at the VQA task. To do this, we propose two different competitor models that will try to outperform the baseline and each other through logical and spatial reasoning.

Our motivation behind these techniques is to build a robust reasoning framework that mirrors human-like logical thinking, thereby addressing the limitations of purely pattern-based approaches.

4.1 Logical Deep Reasoning Model

The Logical Deep Reasoning Model extends the baseline by incorporating a Neural Logical Reasoning Module designed to handle logical operations required for multi-step reasoning. The module reasons over three logical structures `NOT`, `AND`, and `OR`. Each operation is modeled as a transformation of the input question embedding using a fully connected (linear) layer followed by a ReLU non-linear activation function. The motivation for this is to give the model enhanced understanding of the logical relationships between the items in the input question.

Logical Reasoning Module

In this implementation, logical operations (`NOT`, `AND`, `OR`) are applied to the combined image and question features. The module consists of separate fully connected layers for each logical operation, followed by ReLU activations.

- `NOT`: $\mathbf{q}_{\text{NOT}} = \text{ReLU}(\mathbf{W}_{\text{NOT}}\mathbf{q} + \mathbf{b}_{\text{NOT}})$, where $\mathbf{W}_{\text{NOT}} \in \mathbb{R}^{256\times256}$
- `AND`: $\mathbf{q}_{\text{AND}} = \text{ReLU}(\mathbf{W}_{\text{AND}} \cdot \mathbf{c} + \mathbf{b}_{\text{AND}})$, where $\mathbf{c} = [\mathbf{v}'; \mathbf{q}] \in \mathbb{R}^{512}$, $\mathbf{W}_{\text{AND}} \in \mathbb{R}^{256\times512}$
- `OR`: $\mathbf{q}_{\text{OR}} = \text{ReLU}(\mathbf{W}_{\text{OR}} \cdot \mathbf{c} + \mathbf{b}_{\text{OR}})$, where $\mathbf{W}_{\text{OR}} \in \mathbb{R}^{256\times512}$

Integration with Baseline Model:

1. The baseline model uses Resnet-18 [6] to extract **image features** $\mathbf{v}'$ and an LSTM to extract **question text features** $\mathbf{q}$.
2. We apply the `NOT`, `AND`, and `OR` operations to obtain: $\mathbf{q}_{\text{NOT}}$, $\mathbf{q}_{\text{AND}}$, $\mathbf{q}_{\text{OR}}$
3. Merge Reasoning Outputs: $\mathbf{h}_{\text{reasoning}} = \mathbf{q}_{\text{NOT}} + \mathbf{q}_{\text{AND}} + \mathbf{q}_{\text{OR}}$
4. Obtain the prediction by passing $\mathbf{h}_{\text{reasoning}}$ through a fully connected layer with a ReLU activation, followed by a softmax layer.

4.2 Graphical Deep Reasoning Model

We extend the baseline model by adding a **Graph Neural Network (GNN) Module**. The GNN module models relationships between features in images through a graph representation via an adjacency matrix. In this case, the graph was tailored for VQA, where relationships in the graph represent spatial relationships between objects in an image.

Image Encoder and Feature Extraction: We initially tried to utilize the standard ResNet-18 [6] to extract the image features. However, the pooling layers reduced the dimension of the images to a 4×4 space, which would only be valid for a relatively small graph. Thus, we modified some pooling layers in ResNet-18 to increase spatial resolution to 64×64.

Graph Construction

- Feature maps are extracted from a ResNet backbone. These feature maps have dimensions $\mathbf{X} \in \mathbb{R}^{B\times N\times F}$, where:
  - B: Batch size.
  - $N = H' \times W'$: Total number of nodes in the graph, corresponding to the grid elements of the ResNet feature map. H' and W' are the spatial dimensions of the downsampled feature map.
  - $F = C$: Number of features per node, where C is the number of channels in the ResNet feature map.
- A graph is constructed on these N nodes:
  - Each node represents a grid element in the ResNet feature map.

  - An adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ defines the connectivity of the graph. Connections between nodes are based on spatial proximity in the grid (e.g., neighboring nodes are connected).

- A Graph Convolutional Network (GCN) layer is applied to update the node features. The operation is defined as:

$$\mathbf{H} = \text{ReLU}(\mathbf{A}'\mathbf{X}\mathbf{W}),$$

  where:

  - $\mathbf{A}' = \mathbf{A} + \mathbf{I}$: The adjacency matrix with added self-loops ($\mathbf{I}$ is the identity matrix).
  - $\mathbf{W}$: A learnable weight matrix that transforms the features.
  - ReLU: A non-linear activation function applied element-wise.

- The output $\mathbf{H} \in \mathbb{R}^{B \times N \times F'}$ contains updated node features, where F' is the dimensionality of the transformed features.

Integration with Baseline Model:

1. Node features are aggregated via mean pooling to obtain a global representation of the image:

$$\mathbf{h}_{\text{image}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{H}_i,$$

   where $\mathbf{H}_i$ represents the features of node i.

2. The graph image features and LSTM question features are concatenated to form the combined representation:

$$\mathbf{h}_{\text{combined}} = [\mathbf{h}_{\text{image}}; \mathbf{q}],$$

3. To obtain our prediction, we apply on the combined features $\mathbf{h}_{\text{combined}}$ a fully connected layer with a ReLU activation, followed by a softmax layer.

5 Results

We conducted extensive experiments to evaluate the performance of both of these proposed models (Logical and Graphical) compared to the baseline model on the VQA task.

5.1 Training on Full dataset

We first compare the performance of the three models on a full and noise-free dataset. The results are shown in Figure 1.

Hypothesis: we expected that the reasoning models (logical and graphical) would significantly outperform the baseline due to their enhanced capacity for complex reasoning. Furthermore, we expected the logical and graphical models to achieve comparable performance, given their shared design goals but distinct approaches.

Observations: as shown in Figure 1, the enhanced reasoning models outperformed the baseline, achieving a maximum accuracy improvement of approximately $\mathbf{10\%}$. The logical reasoning model slightly outperformed the graphical reasoning model, which can be attributed to the dataset containing a higher proportion of questions requiring logical inference rather than spatial or relational reasoning. This indicates that task-specific reasoning demands play a crucial role in determining model efficacy. Furthermore, we can see that the logical model's **learning speed** is significantly faster than the other two models, with a test accuracy with a 10 epoch advantage.

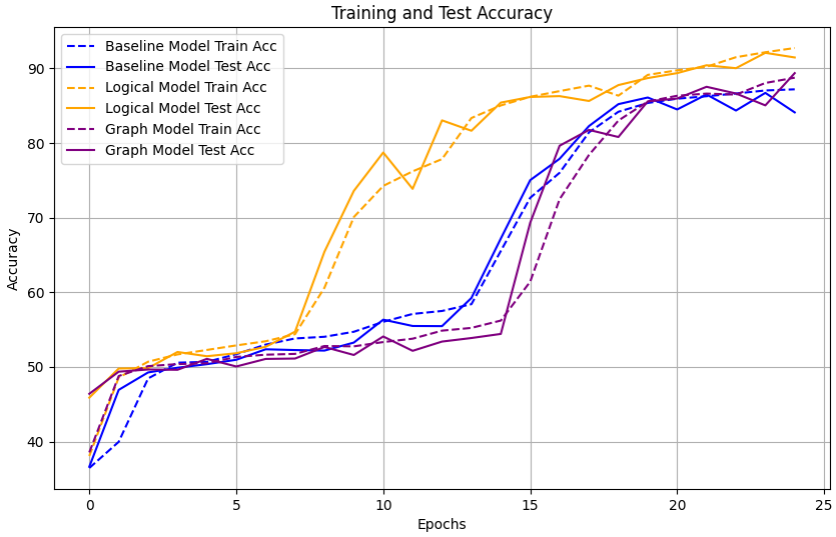

Figure 1: Full Dataset Performance Comparison Of The Three Models

5.2 Learning with Limited Training Data

The goal of this section is to compare how the models fare with access to very limited training data over a larger number of epochs. We progressively reduced the training data to 10%, 20%, and 30% of the original dataset and observed their performance.

5.2.1 Training on 10% of the Dataset

Hypothesis: we expected the logical and graphical reasoning models to outperform the baseline, as reasoning components are better equipped to identify meaningful patterns in limited data scenarios.

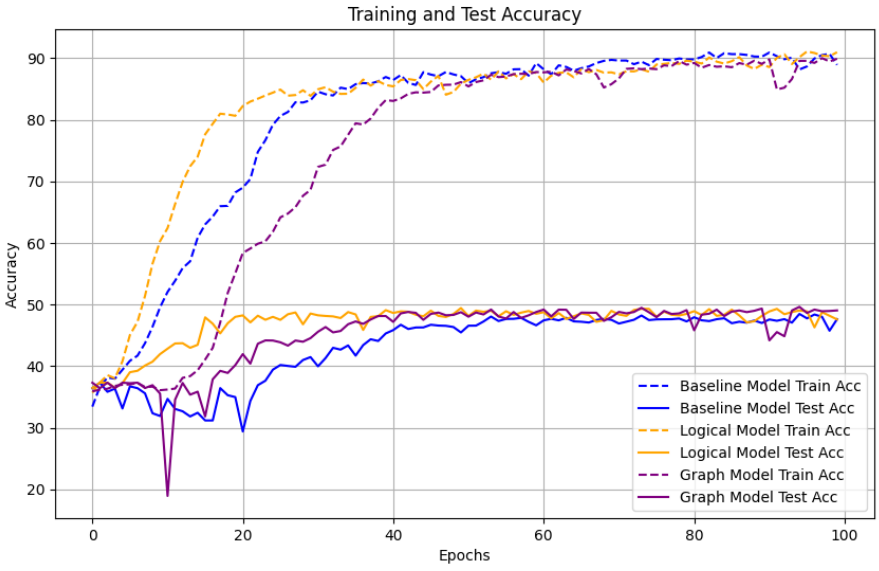

Figure 2: 10% of Dataset Performance Comparison Of The Three Models

Observations: contrary to our expectations, all three models demonstrated similar performance, with no observable advantage for the enhanced reasoning models (Figure 2). This suggests that neither logical nor graphical reasoning modules could effectively leverage patterns in highly scarce data, possibly due to insufficient training signals.

5.2.2 Training on 20% of the Dataset

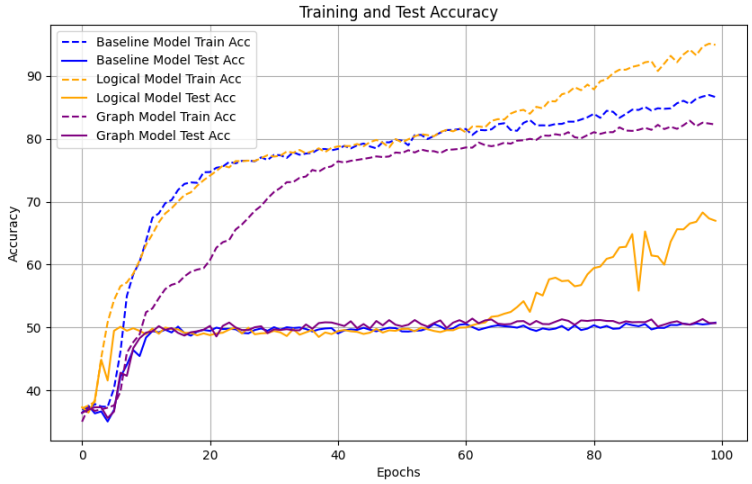

Figure 3: 20% of Dataset Performance Comparison Of The Three Models

Observations: when the training dataset was increased to 20% (Figure 3), a significant divergence emerged. The logical model achieved a substantial improvement, reaching approximately **70% test accuracy**, outperforming both the baseline and graphical models by over **20%**. In contrast, the graphical model and baseline continued to exhibit poor generalization, with learning curves plateauing early. This finding underscores the logical model's capacity to effectively identify patterns and generalize even with limited data.

5.2.3 Training on 30% of the Dataset

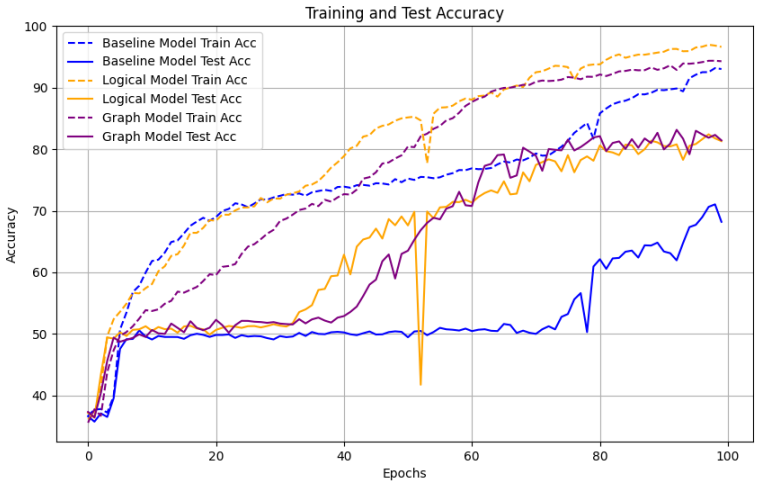

Figure 4: 30% of Dataset Performance Comparison Of The Three Models

Observations: when the training data was further increased to 30% (Figure 4), the graphical model demonstrated significant improvement, catching up to the logical model in test accuracy. While the baseline model also improved, its performance lagged behind, highlighting the advantage of incorporating reasoning modules. This result suggests that the graphical model requires a minimum data threshold (30%) to effectively utilize its spatial reasoning capabilities.

5.3 Robustness to Noisy Data

To evaluate robustness, we introduced noise into the input images by adding Gaussian noise to each pixel:

$$I_{\text{noisy}} = I + \mathcal{N}(\mu, \sigma^2) \tag{1}$$

Where I represents the original image, and $\mathcal{N}(\mu, \sigma^2)$ is Gaussian noise with mean μ and variance σ^2

Hypothesis: we hypothesized that all models would experience a similar degradation in performance, as reasoning models rely on high-quality feature inputs for effective reasoning. Noise in input images would likely obscure critical object or edge information required for both logical and spatial reasoning.

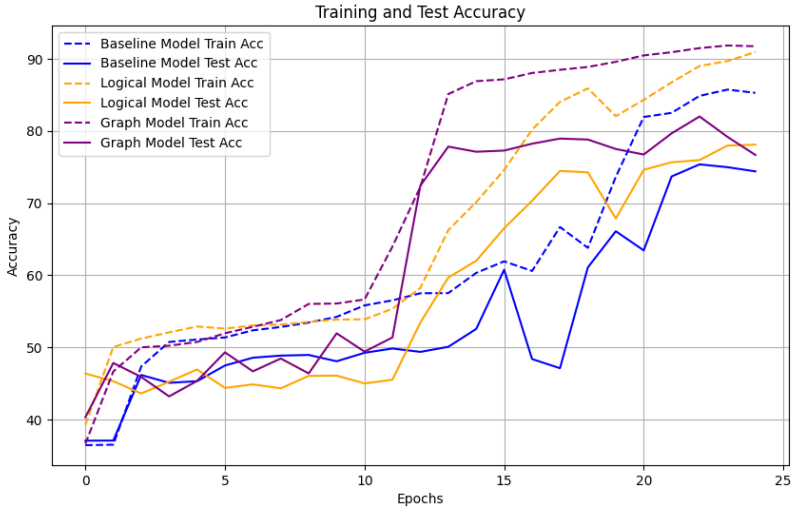

Figure 5: Noisy Dataset Performance Comparison Of The Three Models

Observations: as shown in Figure 5, the performance of all three models was comparable, with test accuracies differing by less than **5%**. However, the logical and graphical reasoning models exhibited a slight advantage over the baseline, suggesting marginal robustness to noisy data. This advantage could arise from their ability to incorporate additional contextual reasoning when input features are degraded.

6 Discussion and Analysis

In this section, we analyze our proposed models' performance, discuss the limitations of our approach, and provide insights into the results obtained. We also suggest potential improvements to address these limitations.

6.1 Analysis of Models and Results

The experimental results demonstrate that integrating reasoning modules into VQA models enhances their performance across various scenarios. The Logical Deep Reasoning Model consistently outperformed the baseline, particularly when trained on limited data and when dealing with complex questions requiring multi-step reasoning.

The Graphical Deep Reasoning Model also showed improvements over the baseline, especially in tasks where spatial relationships and object interactions were crucial. By representing images as graphs and applying GNNs, the model effectively captured relational information that standard CNNs might overlook.

6.2 Insights Gained

- **Importance of Reasoning Modules:** Enhancing deep learning models with reasoning capabilities allows them to better handle tasks requiring logical deductions and multi-step reasoning, which are challenging for standard models.
- **Data Efficiency:** The reasoning modules enable the models to generalize better from limited data, suggesting that incorporating domain knowledge or structured reasoning can reduce the dependency on large datasets.
- **Robustness to Noise:** While noise negatively impacts all models, those with reasoning capabilities are slightly more resilient, potentially due to their ability to focus on essential features and relationships rather than relying solely on pattern recognition.

6.3 Limitations of Our Approach

Despite the promising results, our approach has several limitations: our models assume that the dataset consists of clear images with distinguishable geometric shapes and that the questions are well-structured. In real-world scenarios, images may contain noise, occlusions, or complex backgrounds, and questions may be more ambiguous. Secondly, incorporating GNNs increases computational requirements due to the need to construct and process graphs for each image. This may not scale well to larger datasets or images with higher resolutions. Lastly, and most importantly, our models were tested on a dataset with simple geometric shapes. Their performance on more complex and diverse datasets, such as those involving real-world images with multiple objects and intricate relationships, remains uncertain.

6.4 Potential Improvements and Future work

To address the limitations and further enhance our models, we propose several improvements. We would add a natural language processing (NLP) component to parse questions and extract logical structures explicitly to allow the Logical Deep Reasoning Model to apply logical operations more effectively. To mitigate computational complexity, techniques like graph sampling, sparse representations, or using more efficient GNN architectures could be employed to improve scalability. But most importantly, we would have to evaluate the models on much more complex and diverse datasets, such as CLEVR [12] or VQA v2 [13]. This would provide insights into their generalization capabilities and help identify areas for further refinement.

References

[1] Kahneman, D. (2011). *Thinking, Fast and Slow*. Farrar, Straus and Giroux.

[2] Zhou, V. (2021). *Easy-VQA: An Easy-to-Use Visual Question Answering Dataset*. Retrieved from: https://github.com/vzhou842/easy-VQA/tree/master

[3] Chen, D., Bai, Y., Zhao, W., Ament, S., Gregoire, J. M., & Gomes, C. P. (2019). *Deep Reasoning Networks: Thinking Fast and Slow*. arXiv preprint arXiv:1906.00857.

[4] Anthony, T., Tian, Z., & Barber, D. (2017). *Thinking Fast and Slow with Deep Learning and Tree Search*. arXiv preprint arXiv:1705.08439.

[5] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2008). The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1), 6180.

[6] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770778).

[7] Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (pp. 15321543).

[8] Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., & Dahl, G. E. (2017). Neural Message Passing for Quantum Chemistry. In *Proceedings of the 34th International Conference on Machine Learning* (pp. 12631272).

[9] Kipf, T. N., & Welling, M. (2016). Semi-Supervised Classification with Graph Convolutional Networks. arXiv preprint arXiv:1609.02907.

[10] Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. arXiv preprint arXiv:1412.6980.

[11] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems* (pp. 80268037).

[12] Johnson, J., Hariharan, B., van der Maaten, L., Hoffman, J., Fei-Fei, L., Zitnick, C. L., & Girshick, R. (2017). CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2901–2910.

[13] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., & Parikh, D. (2017). Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6904–6913.

7 Appendix

7.1 Training on Full Dataset

Figure 6: Baseline Model Training and Test Loss

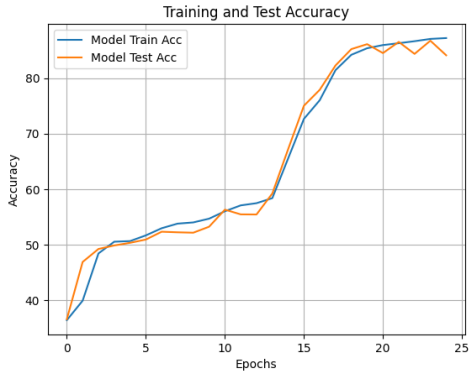

Figure 7: Baseline Model Training and Test Accuracy

Figure 8: Logical Model Training and Test Loss

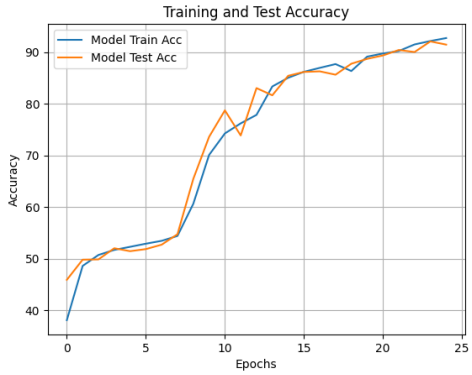

Figure 9: Logical Model Training and Test Accuracy

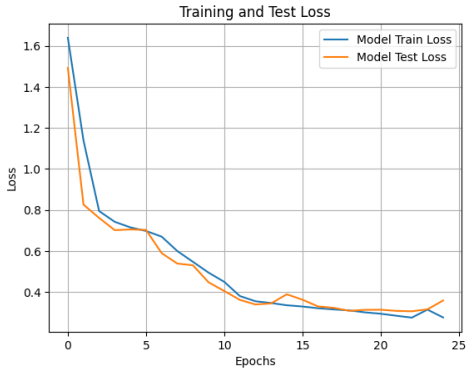

Figure 10: Graph Model Training and Test Loss

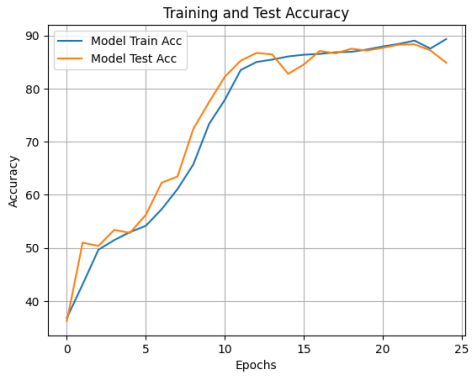

Figure 11: Graph Model Training and Test Accuracy

7.2 Robustness to Noisy Data

Figure 12: Baseline Model Training and Test Loss

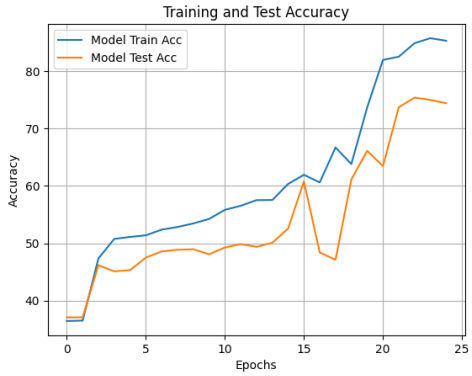

Figure 13: Baseline Model Training and Test Accuracy

Figure 14: Logical Model Training and Test Loss

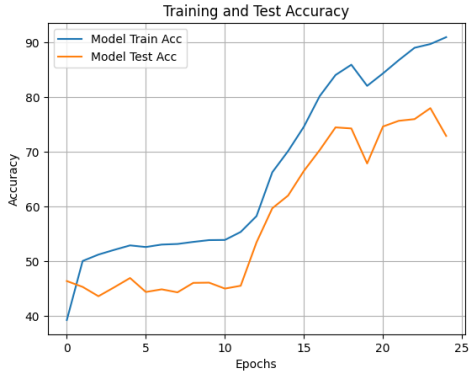

Figure 15: Logical Model Training and Test Accuracy

Figure 16: Graph Model Training and Test Loss

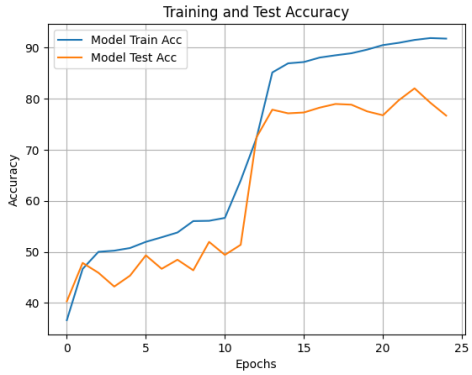

Figure 17: Graph Model Training and Test Accuracy

7.3 Learning with Limited Training Data

7.3.1 Learning on 10% of the dataset

Figure 18: Baseline Model Training and Test Loss

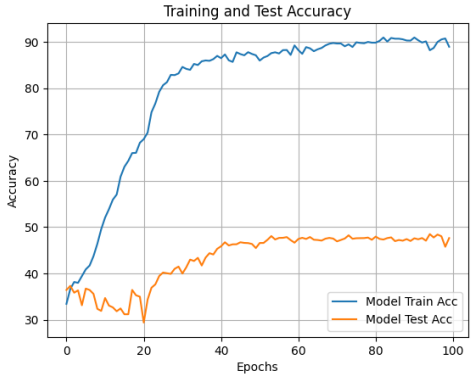

Figure 19: Baseline Model Training and Test Accuracy

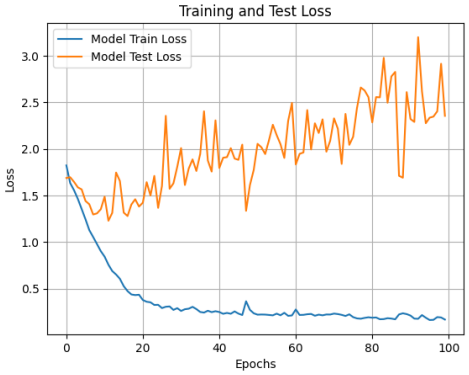

Figure 20: Logical Model Training and Test Loss

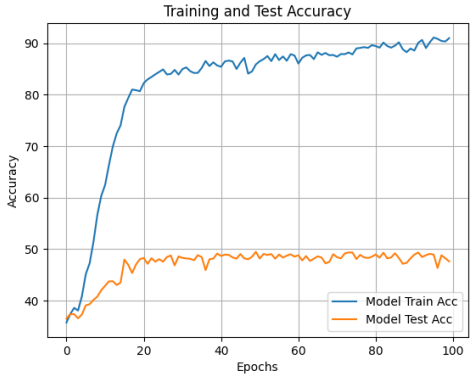

Figure 21: Logical Model Training and Test Accuracy

Figure 22: Graph Model Training and Test Loss

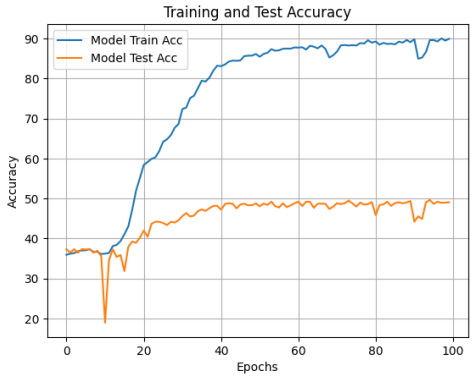

Figure 23: Graph Model Training and Test Accuracy

7.3.2 Learning on 20% of the dataset

Figure 24: Baseline Model Training and Test Loss

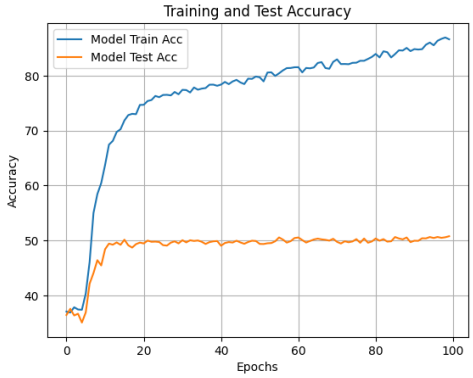

Figure 25: Baseline Model Training and Test Accuracy

Figure 26: Logical Model Training and Test Loss

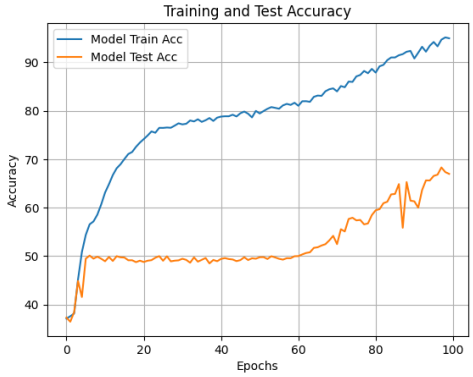

Figure 27: Logical Model Training and Test Accuracy

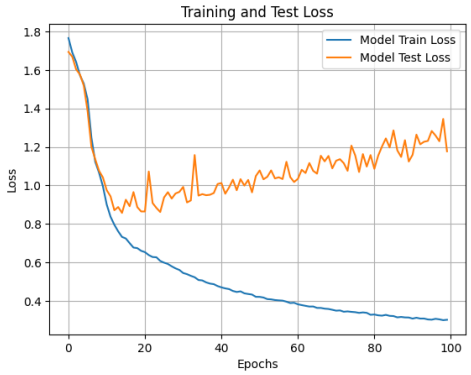

Figure 28: Graph Model Training and Test Loss

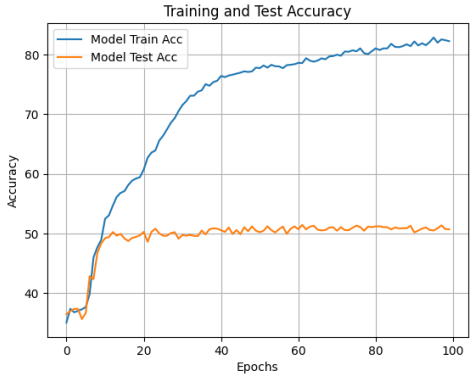

Figure 29: Graph Model Training and Test Accuracy

7.3.3 Learning on 30% of the dataset

Figure 30: Baseline Model Training and Test Loss

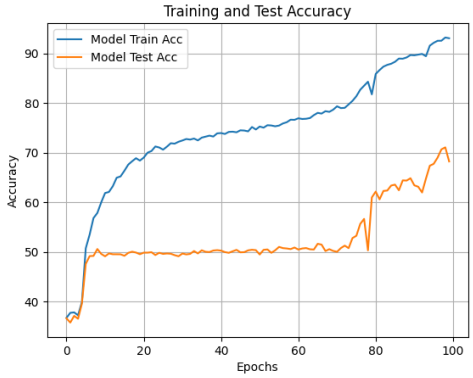

Figure 31: Baseline Model Training and Test Accuracy

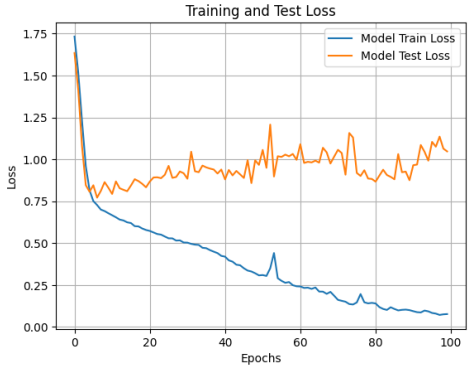

Figure 32: Logical Model Training and Test Loss

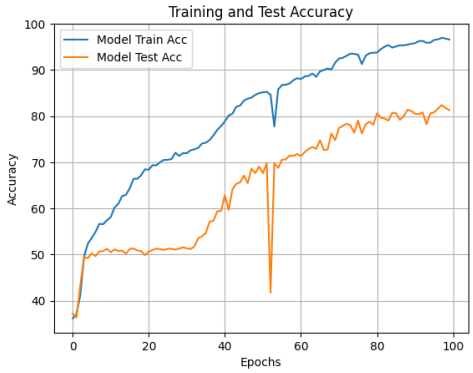

Figure 33: Logical Model Training and Test Accuracy

Figure 34: Graph Model Training and Test Loss

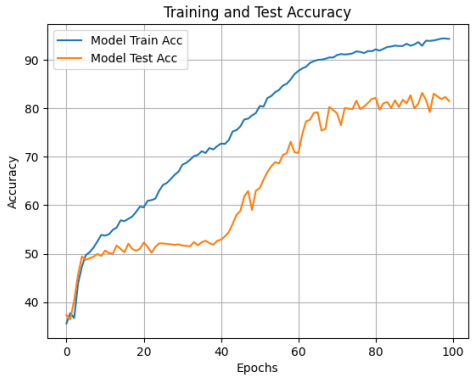

Figure 35: Graph Model Training and Test Accuracy