Graph-Structured Representations for Visual Question Answering

Chaonan Song

May 11, 2018

1 Introduction

Language representation: Because of the above reasons, they use the extensive existing work in the natural language community to help solve problems. First, they use the reader to identify the grammatical structure of the problem [2]. This produces a graphical representation where each node represents a word and each side represents a specific type. Then they associate each word (node) with a vector pre-trained on a large text data set [3]. This association maps words to semantically meaningful spaces. So, this basically regulates the rest of the network to share learning concepts in related words and synonyms. This is helpful for dealing with rare words, and allows problems including lack of words in training questions/answers. It should be noted that this pre-training and temporary processing of the language part imitates the exercises commonly used in the image part, where the visual features are usually obtained from a fixed CNN, which itself is pre-trained on a larger data set and has different goals.

Scene representation: Each object in the scene corresponds to a node in the scene graph that has associated feature vectors that describe its appearance. The graphics are fully connected, with each edge representing the relative position of the object in the image.

Applying Neural Networks to graphs: The advantage of this method of using text and scene graphs without using typical representations is that graphs can capture relationships between words and objects with semantic importance. This enables GNN to exploit (1) the disordered nature of the scene (especially the object) and (2) the semantic relationship between the elements. This is in contrast to the method of using CNN to represent the image and the processing word to continuously process the problem using RNN (though the grammatical structure is very non-linear). The graph representation ignores the order in which elements are processed, but instead represents the relationships between different elements using different edge types. Their network uses multi-level iterations to traverse features associated with each node, ultimately identifying soft matches between nodes in the two graphs. This match reflects the correspondence between the words in the question and the objects in the image. The characteristics of the

matching node are fed into the classifier to infer the answer to the problem (Figure 1).

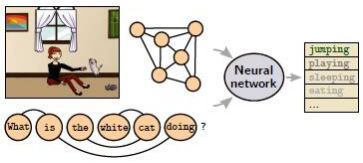

Figure 1: We encode the input scene as a graph representing the objects and their spatial arrangement, and the input question as a graph representing words and their syntactic dependencies.

The main contributions of this paper are shown in Table 1.

Recall rate at 100 FP on FDDB			
They describe how to use graph representations of scene and question for VQA.	They showed how to use off-the-shelf language parsing tools to generate graphical representations of grammatical relationships.	They trained the proposed model on the VQA abstract scene benchmark [1] and improved accuracy and accuracy.	They assessed the uncertainty in the model by first presenting the VQA task-predicted answer precision memory curve.

Table 1: Contribution of this article

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. *In Proc. IEEE Int. Conf. Comp. Vis*, 2015.

[2] Marie Catherine De Marneffe and Christopher D. Manning. The stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8, 2008.

[3] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543, 2014.