Extending MAC Network for Visual Reasoning with GQA

Zora Che
Boston University
zche@bu.edu

Abstract

The complex reasoning challenge posed by the linguistically diverse GQA dataset raises the question of how machine reasoning should be approached. In opposition to large cross-modal transformer models such as UNITER focused on learning shared representations, the MAC Network is an example of an attention-based multimodal model focused on sequential reasoning. This paper explores extending the MAC model via pretrained features, merging spatial and object features, and projecting control steps onto intermediate memory. The paper discusses the challenges of training on a subset of the GQA dataset, as well as directions for increasing the capacity for sequential reasoning with MAC. The augmented model with object features merged with spatial features improved the best baseline model by 1.4%. Though challenge remains in extending the network for unseen images for question answering pairs.

1. Introduction

Question answering has been a coveted focus for understanding machine reasoning capabilities, as well as building towards trustworthy human-computer interaction. Despite state-of-the-arts results in vision and language respectively, the multimodal problem of question answering has not seen a dramatic increase in accuracy among other metrics in recent years. Other than synthetic datasets such as CLEVR, model performances on real data for VQA (Visual Question Answering), VCR (Visual Commonsense Reasoning), etc. have stagnated for the last few years. Moreover, many critique that popular datasets such as VQA are problematic as benchmarks for machine reasoning. Drew Hudson and Christopher Mannings argue that most visual question answering benchmarks are limited in exhibiting: strong language and real-world biases, such that models may make educated guesses; visual biases such that most objects of interest are also the most salient; unclear error source, such that there is lack of object grounding; illogical and inconsistent answering; as well as little reasoning or compositionality. The worry for ill-posed datasets and subsequent benchmarks is that models may be drawing on prior dependencies and relationships, rather than reasoning over the given question and answer.

Thus, Drew Hudson and Christopher Mannings introduced the GQA dataset designed for complex compositional question answering over real-world, balanced images. The dataset draws on a diverse set of reasoning skills, with semantics included for dataset questions as well. The questions are generated using a rule-based multi-step question engine focused on linguistic diversity and a large vocabulary. In addition, they provided a set of metrics exploring issues with model behaviors such as inconsistency as mentioned above. The complexity of reasoning in this new dataset is illustrated through the stunted performance of state-of-the-art VQA models on GQA.

The team behind GQA also introduced the MAC Network for step-based reasoning aimed at tackling compositionality. MAC has demonstrated its ability to excel in complex reasoning tasks on synthetic datasets, and its baseline outperforms models relying on generic LSTM and CNN implementations for visual question answering by 10% [1]. Considering MAC's high performance on synthetic datasets, it is of interest to extend its architecture in search of evidence to support a step-based model for generalized visual reasoning.

2. Background

2.1. MAC Network

The MAC Network is a fully differentiable attention-based recurrent network designed to aid compositional reasoning with inspirations from computer architecture [1]. The original paper proposes visual reasoning in steps for compositional reasoning and generalizing complex image relationships and semantic structures. The network achieved 98.9% accuracy on CLEVR, a synthetic and balanced dataset testing challenging reasoning skills, such as transitive and logical relations, counting, and comparisons.

In the input cell, the network relies on a differentiable bidirectional LSTM model to encode the question. The embedded question and the processed image are queried in steps in the MAC cells via attention based procedures. The model is linked by a number of MAC cells, and in each cell the control, read, and write units work together to update internal memory which eventually lead to the output cell where a classification is made over all available answers or vocab words in the case of the GQA dataset.

The control unit represents the decision making at step i. The question is selected at time i via a learned matrix transformation representing aspects of the question, q_i relevant to the i^{th} step. Following the initialized control c_0, each subsequent control is produced by first combining c_{i-1} and q_i through a linear transformation, $cq_i = W[c_{i-1}, q_i] + b$, then casting it to the question space. Similarity is measured between cq_i and each question word cw_s, by computing the Hadamard product between the two and transforming it with a trainable matrix $ca_{i,s} = W(cq_i \odot cw_s) + b$. Then the result passes through softmax for generating a distribution, $cv_{i,s} = softmax(ca_{i,s})$. The words are then summed based on this distribution to produce the next control, $c_i = \sum_s cv_{i,s} \cdot cw_s$.

The read unit takes instruction from the control, and retrieves information from the knowledge base $k_{h,w}$ conditioned on previous memory m_{i-1} and current control. It first compute interaction between knowledge base and the last memory, $I_{i,h,w} = [Wm_{i-1} + b] \odot [Wm_k h, w + b]$. Another intermediate memory is calculated by concatenating element of the knowledge base and current interaction and transforming through learnable matrix, in hopes of consider information not directly related to previous result, $I'_{i,h,w} = W[I_{i,h,w}, k_{h,w} + b]$. Another attention mechanism is then performed, similar from the one in control, where we compute the similarity between control and intermediate states, $ra_{i,h,w} = W(c_i \odot I'_{i,h,w}) + b$. Thus producing attention distribution over softmax, $rv_{i,h,w} = softmax(ra_{i,h,w})$. This is used to compute a weighted average with regard to elements in the knowledge base, $r_i = \sum_{h,w} rv_{i,h,w} \cdot k_{h,w}$.

The write unit combines the new retrieved information r_i with prior intermediate result m_{i-1} by linear transformation to create the current memory state, $m_i = W[r_i, m_{i-1}] + b$.

The output takes in the final memory and the question, concatenating and transforming, leading to a result via a 2-layer fully-connected softmax classifier over all possible answers.

The MAC model utilizes attention iteratively in its information retrieval both from the questions and the knowledge base, yet the two attention mechanisms are strictly separated, and the author believes that it leads to better generalizability and interpretability. The authors propose that separating the attention mechanisms encourages reasoning based on combining two modalities via step-specific reasoning procedures, rather than exploiting dependencies between text and visual similarities.

2.2. Attention

Attention mechanisms encourage models to focus on important parts of visual or language inputs at each step of the task. In recent years, attention mechanisms, especially when trained in parallel, have pushed the state-of-the-arts in natural language processing, as well as image classification [2]. Textual attention, such as the one employed in transformer models like BERT, find semantic alignments within an encoder decoder framework for revealing long-term dependency.

Applying attention over the images has been heavily explored for visual reasoning, such as the unimodal approach of stacked attention networks which perform multi-step visual attention proposed by Yang et. al [3]. Moreover, dual attention is proposed by worked by Nam et. al [4], where model applies both visual and textual attentions, and refines both attentions via multiple reasoning steps based on the memory of previous attentions, facilitating close interplay with visual and textual data.

3. Extension and Implementation

3.1. Data Selection

The GQA dataset provides raw data of question image pairs, as well as scene graphs where the questions are generated from, and preprocessed image features. The balanced datasets from official released are balanced in terms of linguistic diversity distribution, as well as separation of images between train, valid, and test set. The goal of segregating images was to encourage generalizable reasoning from previously unseen images. However, when training from a smaller subset of the entire dataset, training, validating, and testing on unseen images because a few shot or zero shot challenge based on image and question features alone. Given the wide varieties of linguistic queries, training on the subset limits the model's ability to extrapolate abstract relations and concepts, since only a limited of words, relations, and query types are included. Considering the long tail distribution of natural language, and the diversity within the GQA dataset, extrapolating to unseen words and images when training on a subset proved to be difficult.

I first trained on the official segmentation of balanced data, selecting 200000 from the training set, 40000 from the validation set, and the entire testdev set with 12578 questions. Despite baseline and augmentations models all surpassing 40% in training and validation accuracy, testing on never seen images with new questions and answers lowered accuracy to below 5%, and less than 1% for the baseline model with object features. The result is not unsurprising since many of the answers to the test set are out of vocabulary for the model to begin with, and the model did not have some of the answers as candidates. Surpassing limitations in vocabulary and image generality may be ameliorated through diversifying and increasing training dataset size, and/or employing pretrained networks. However due to the computational constraint and consideration for project scope, it was not feasible for me to train on the full GQA dataset or on other datasets such as Visual Genome.

For a smaller scope, I chose to retrain all my models and augmentations on a new subset of the data, abandoning the strict separation of images between train, validation, and testing. I considered this a realistic goal for the project since predicting on questions with never seen vocabulary, and analyzing never seen objects are in themselves open unimodal questions. Though the learned reasoning is more data specific due to our setup, I wish to argue that such dependency is inevitable as current state-of-the-art models often require large knowledge bases to provide missing data-specific insights. Conceptual reasoning still requires considerate knowledge grounding.

Moreover, augmenting upon the baseline performance of the segregated dataset is possible but empirically inapplicable. Since the baseline was below 1% to start with, marginal improvements or lack of improvements are difficult to analyze, since only a few random correct answers may change the accuracy.

Therefore, I chose to train and test on a smaller subset of questions sharing a set of images. I permutated the first 200000 entries in the balanced train JSON file, getting 75%, 15%, 15% split for train, validation, and testing respectively. There are 660099 question answer pairs for training set, 141838 question answer pairs for validation set, and 141063 question answer pairs for testing set.

Through the random permutation process, each set still retains diversity in questions as well as images. Since the images are not evenly split among the sets, the processed test question images pairs still have some that are never seen images from the train/valid set. As such, the challenge of generalizing based on few or no shots still exist though to a lesser extent.

3.2. Preprocessing

The author proposed processing questions with NLTK tokens then embedding them with GloVe when training/testing. In hope of introducing more linguistic diversity in terms of richer feature representations of the given questions, I also preprocessed the question with the BERT model after tokening following BERT conventions, generating features from BERT that were used to replace the bidirectional LSTM. The BERT outputs have shape $[30, 768]$.

I trained with preprocessed GQA image features, both object based and spatial-based. The object-based features were exacted using fast R-CNN detectors by dataset authors, with the shape $[100, 2048]$, where each row contains the feature with shape 2048 for one object. Though an image may have up to 100 object features, most do not. The spatial features were extracted by Resnet-101, with the shape $[2048, 7, 7]$.

3.3. Baseline Implementation

I implemented the baseline in PyTorch, building upon the opensource project by Ronil Pancholia [1]. I tuned the baseline model via three main parameters: max step of MAC cells, learning rate, and epochs. I defaulted to the dropout rates in the official Tensorflow based repository. I found that performance stagnates past over 5 MAC cells, and that some of my trials for max steps 5 and 10 had the same accuracy. This may infer the model can reason the question in around 5 steps as well as around 10 steps. Considering the average length of questions in my training set is under 10 words, setting a lower max cell eases training and has better results. Training and validation accuracy for epochs over 10 also stagnates, and the model performs better at 10 epoch. The best baseline model was trained under using 5 MAC cells, a learning rate of 0.0001, 10 epochs, and training with the spatial features. The accuracy of 55.2% is around 10% less than the unofficial best MAC model results on GQA achieved by the authors mentioned in presentations, though the difference is expected due to the difference in data size. The best model using object features is 53.4% with the same parameters. The object features were pretrained on images/scene graphs in the GQA training set, using a bottom-up-attention paradigm. However it still seems that object features would require scene graph information in conjunction in training to bypass limits in compositional reasoning.

3.4. Extending with Pretrained Features

In an attempt to introduce pretrained dependencies for complex questions, I replacing the LSTM layer with pretrained BERT features, transforming the BERT output to the expected dimension as the input question, and transforming the BERT state to be the initialized control for MAC. Both transformations are implemented via trainable linear layers.

3.5. Merging Object and Spatial Features

In baseline, spatial features had higher accuracy than object features in training, validation, and testing. To increase the information accessible to the model at each read stage, two types of merging were experimented on: a transformation of the object features with Hadamard product, and via applying the attention between the object and spatial features as a mask for the spatial features.

For the first method, a trainable linear layer is applied on object features to project it into the same shape as spatial features, via haddamard product, the combined image is then used as the knowledge base $h = W[O] + b$; $k = h \odot S$,

[1] https://github.com/ronilp/mac-network-pytorch-gqa

where O and S are resized object and spatial features respectively, and k represents the knowledge base.

For the second method, an attention map is created by the attention module. The spatial features are convolved to produce features of size $[512, 7, 7]$, and object features are linearly transformed twice with a dropout of 0.5 for the first layer to generate object features of size $[512]$. This feature vector is then tiled over all spatial positions in the resized spatial feature. The resulting mask is applied via addition then passed through ReLU, lastly the mask is convolved to the dimension $[2, 7, 7]$, where 2 represents two glimpses of the attention mask. Since we are operating with soft attention, we choose to apply the mask twice, indicated by the glimpse number. The cross-modal attention is then applied to the spatial features, which is then used as the knowledge base.

The attention module is employed with the hope of generating attention masks that may inform which spatial features, and/or which spacial feature in connection are notable via coattention between the spatial features and object features.

3.6. Projecting Control onto Internal Memory

The original paper made the distinction that the separation between control and internal memory is not only good for visualization and demonstration of interpretability, but also good for generalizing performance. Consider that the mechanism in MAC is broadly similar to the method in Dual Attention Network by Nam et. al [4] in that the question is attended to sequentially via attention, producing c_i, and the image is also attended to sequentially via attention, producing r_i. In the Dual Attention Network, attention is applied for each modality individually, then combined together with the prior memory, as such $m_i = m_{i-1} + v_i \odot u_i$ where v_i and u_i are the respective modality after applying attention. Taking inspiration from DAN, I tested projecting control onto internal memories, for it to be combined into the next memory, $m_i = W[r_i \odot c_i, m_{i-1}] + b$. In addition to this method, I tested transforming the control first via a linear layer and softmax, then projecting it via Hadamard product with prior memory, $h_i = softmax(W[ci] + b)$; $m_i = W[r_i \odot h_i, m_{i-1}] + b$.

4. Results

The model is evaluated on accuracy for the output answers, where 1 for correct answers and 0 for incorrect answers to calculate the percentage of correct answers on the testing set.

4.1. Extending with Pretrained Features

Training with BERT features resulted in the lowest performance across the board, in training, validation, and testing. The best model using BERT is 10% less than basline implementation with LSTM. This supports that a fully differentiable model may tune to parts of the question better than linearly connecting pretrained features. Considering that the pretrained feature also may have a mismatch between the vocabulary in the subset of the dataset compared to the vocabulary in the pretrained network. Moreover, the uncased BERT was trained via masked language modeling, which aids in classification and next sentence prediction, though in order to perform well for the GQA dataset, the model needs to have refined distinctions between types of questions.

The results here suggest that a fully differentiable network receiving tokenized questions is more adaptable to the semantic space of a given dataset, especially if the task on hand requires refined question analysis. An improvement to this method would be to include a pretrained BERT network as the input layer, with increased computational and training time demands. This augmentation is denoted in the table as +BERT.

4.2. Merging Object and Spatial Features

Computing elementwise multiplication between spatial features and transformed object features produced the best overall augmentation accuracy of 56.6%. The result may allude to possible matching the network learns between the two different views of the image modality. It agrees with the assumption that the MAC Network improves with more complete information and more knowledge. This augmentation is denoted in the table as Spatial+Object $\odot$.

The attention module approach achieves an accuracy of 55.0% for 5 steps and 10 epochs, similar to the baseline result with one knowledge source. Fine tuning to lower max steps further showed decreased results. The lack of improvement with the attention module is likely due to the difficulty to optimize as the two views are attended to at the input cell only, instead of being iteratively refined through each recurrent cell. Moreover, both the spatial and object features were scaled down due to computational limits, which could cause the combined image to lose information. Considering that the attention module is parameter intensive, it is possible that more varied data may aid in finetuning this model. This augmentation is denoted in the table as Spatial+Object (attention).

4.3. Projecting Control onto Internal Memory

Projecting the control onto the internal memory by elementwise multiplication decreased accuracy slighly, denoted as ”mul” in the results table. This could suggest that separating the two attention processes may avoid overfitting or learning shared occurrences instead of reasoning. The critique for optimizing for combined visual and language attention is that for complex questions, shared similarities are not enough.

Max 5 Steps	Accuracy (10 epochs)
Baseline (Object)	0.534
Basline (Spatial)	0.552
Object+mul	0.518
Object+BERT	0.430
Spatial+mul	0.552
Spatial+FN	0.545
Spatial+Object ($\odot$)	0.566
Spatial+Object (attention)	0.550
Spatial+Object+FN	0.564
Spatial+BERT	0.468

Table 1. Results on test set with 5 MAC cells.

Max 10 Steps	Accuracy (10 epochs)
Baseline (Object)	0.533
Basline (Spatial)	0.557
Object+BERT	0.432
Spatial+mul	0.552
Spatial+FN	0.545
Spatial+Object ($\odot$)+FN	0.508
Spatial+BERT	0.451

Table 2. Results on test set with 10 MAC cells.

Max 10 Steps	Accuracy (20 epochs)
Basline (Spatial)	0.559
Spatial+FN	0.546

Table 3. Results on test set with 5 MAC cells and 20 epochs.

The other method of projecting the control via a softmax activated feed forward layer also decreased accuracy slightly. This denoted as ”FN” in the results table.

5. Discussion and Future Work

This paper explores extending the MAC model and the limits to sequential reasoning models. The results here demonstrate insight on a subset of the GQA dataset, and further work is needed to demonstrate the benefits or drawbacks of augmentations, especially for the attention module with a large set of parameters that may perform better on a larger training set with increased diversity. We showed that increasing the diversity of the information by combining spatial and object features for the knowledge base contributed to an increase in MAC performance. We also showed that the separation between the internal attention-based representation of question and image through the separation of control and memory seems to allow the network to focus on reasoning generally without having to ground visual and language features together. Further work should also explore extending pretrained models for trainable parameters and additional generalizability for adapting to varied real-world questions and images.

References

[1] Hudson, D. A., Manning, C. D. (2018, February). Compositional Attention Networks for Machine Reasoning. In International Conference on Learning Representations.

[2] Shih, K. J., Singh, S., Hoiem, D. (2016). Where to look: Focus regions for visual question answering. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4613-4621).

[3] Yang, Z., He, X., Gao, J., Deng, L., Smola, A. (2016). Stacked attention networks for image question answering. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 21-29).

[4] Nam, H., Ha, J. W., Kim, J. (2017). Dual attention networks for multimodal reasoning and matching. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 299-307).