[p3]

Problem being solved

Generally, VQA model can be classified by its structure, which is monolithic network and modular network.

* This paper presents the first Neural Module Networks (NMNs), which specifies a framework for modular, composable, jointly trained neural networks. A network model is tailored to each question in the VQA dataset. This means that the network of the NMN model is dynamically generated according to the linguistic structure of the QUESTION.

[p4]

First, state-of-the-art performance on the full range of computer vision tasks that are studied requires a variety of different deep network topologies—there is no single “best network” for all tasks. Second, though different networks are used for different purposes, it is commonplace to initialize systems for many of vision tasks with a prefix of a network trained for classification.

It is not hard to think about, Can we generalize this idea in a way that is useful for question answering? Since the more challengable, the more valuable. Rather than thinking of question answering as a problem of learning a single function to map from questions and contexts to answers, it’s perhaps useful to think of it as a highly-multitask learning setting, where each problem instance is associated with a novel task, and the identity of that task is expressed only noisily in language.

* This type of approach considers the question as a combination of a series of base modules (e.g. find, relate, count, etc.) whose functions can be fitted with sub-networks, and you need to select different modules to answer different questions, so the network structure is problem-dependent and dynamic. Therefore, this kind of flexible networks are more intuitive and interpretable than giant networks, and the process is also more transparent.

[p6]

Step1: Analyse each question using the semantic parser and combine the analysis to obtain the module layout (including the basic computational modules needed to answer the question and the relationships between them).

Step2: Combine to generate task-specific modules that answer the questions. The modules need to be manually designed internally and the information passed between modules may be raw image features, attention, or classification decisions. All modules in the NMN are independent and combinable, which makes the computation different for each problem instance and may not be observed during training.

The figure starts by generating an attention to the dog (the attend module), which passes its output to a location classifier (the classify module).

Step3: The final answer uses a recurrent network (LSTM) to read the problem input and combines it with the output of the NMN to obtain the classification result.

[p7]

The modules operate on three basic data types: images, unnormalized attentions, and labels.

Form: module names are typeset in a fixed width font, like this TYPE[INSTANCE](ARG1, . . . ). TYPE is a high-level module type of the kind described in this section. INSTANCE is the particular instance of the model under consideration—for example, attend[red] locates red things, while attend[dog] locates dogs. Weights may be shared at both the type and instance level. Modules with no arguments implicitly take the image as input; higher-level arguments may also inspect the image.

[p8]

Module type:

左上：attention module: attend[] convolves every position in the input image with a weight vector to produce a heatmap or unnormalized attention.

右上：A re-attention module re-attend[c] is essentially just a multilayer perceptron with (ReLUs), performing a fully-connected mapping from one attention to another. So re-attend[above] should take an attention and shift the regions of greatest activation upward while re-attend[not] should move attention away from the active regions.

左下：A combination module combine[c] merges two attentions into a single attention. For example, combine[and] should be active only in the regions that are active in both inputs, while combine[except] should be active where the first input is active and the second is inactive.

右下：A classification module classify[c] takes an attention and the input image and maps them to a distribution over labels. For example, classify[color] should return a distribution over colors in the region attended to.

中：A measurement module measure[c] takes an attention alone and maps it to a distribution over labels. Because attentions passed between modules are unnormalized, measure is suitable for evaluating the existence of a detected object, or counting sets of objects.

[p9]

Parsing: Parsing each question with the Stanford Parser to obtain a universal dependency representation. Dependency parses express grammatical relations between parts of a sentence (e.g. between objects and their attributes, or events and their participants), and provide a lightweight abstraction away from the surface form of the sentence. The parser also performs basic lemmatization, for example turning kites into kite and were into be. This reduces sparsity of module instances.

* **Layout:** Based on specific tasks, converts symbolic representations into modular network structure.

In addition to providing sentences to get the module network via parser and layout, it is also possible to provide direct sql-like query statements that specify the requirements precisely

[p10] The LSTM network allows us to model the underlying syntactic rules in the data. Secondly, it allows us to capture semantic patterns. For example, for questions: what is flying and what are flying, is and are will both be converted into be, so both get converted to what(fly).

Text

Description automatically generated

Text

Description automatically generated

The question modeling component predicts a distribution over the set of answers, like the root module of the NMN. The final prediction from the model is a geometric average of these two probability distributions, dynamically reweighted using both text and image features. The complete model, including both the NMN and sequence modeling component, is trained jointly.

[p11]

Graphical user interface, text

Description automatically generated

[p14]

In this paradigm, designers have access to standard neural parts from which to construct models for performing complex reasoning tasks. While visual question answering provides a natural test platform for this approach, its functionality is potentially much broader, extending to queries about documents and structured knowledge bases or more general signal processing and function approximation.