

视觉问答 Visual Question Answering (VQA) 简单综述

WANNG: 文本主要是对VQA整个任务做一个综述。信息来源包括但不限于以下论文等。

大部分来源于以下材料：

- Visual Question Answering using Deep Learning: A Survey and Performance Analysis [1]
- Visual Question Answering: A Survey of Methods and Datasets [2]
- Survey of Visual Question Answering: Datasets and Techniques [3]
- 视觉问答-1_综述.md [4]
- Visual Question Answering: Datasets, Algorithms, and Future Challenges [5]

目录 content

- 0. 本文结构
- 1. VQA任务简介
 - VQA和NLP任务的区别
 - VQA和CV任务的区别
 - 基于对象检测的任务
 - 图像描述任务
- 2. VQA 相关的数据集 (datasets)
 - DAQUAR
 - COCO-QA
 - VQA数据集
 - VQA v1
 - VQA v2
 - Visual Madlibs
 - Visual Genome
 - Visual7W
 - CLEVR
 - Tally-QA
 - KVQA
- 3. 主流模型与方法
 - 3.1 非深度学习方法
 - 回答类型预测 Answer Type Prediction (ATP)
 - 多元世界问答 Multi-World QA
 - 3.2 无注意机制的深度学习模型 Non-attention Deep Learning Models

- iBOWING
- Full-CNN
- 神经元询问 Ask Your Neurons (AYN)
- Vis + LSTM
- Vanilla VQA (deeper LSTM Q + norm I)
- 动态参数预测 Dynamic Parameter Prediction (DPPnet)
- 3.3 基于 Attention 的模型 Attention Based Models
 - Where To Look (WTL)
 - 循环空间注意 Recurrent Spatial Attention (R-SA)
 - 堆叠注意网络 Stacked Attention Networks (SAN)
 - 层次协同注意 Hierarchical Co-Attention model
 - 双重注意网络 DAN
 - Tips and Tricks for Visual Question Answering
 - Pythia v0.1
 - Focal Visual-Text Attention (FVTA)
- 3.4 其它有趣的模型
 - MCBP for VQA
 - 神经模块网络 Neural Module Network (NMN)
 - AMA based on KB
 - NS-VQA
 - 差分网络 Differential Networks
- 3.5 暂时的小结
- 3.6 基于 Transformer (BERT) 的模型 Transformer Based Models
 - ViLBERT
 - VisualBERT
 - LXMERT
 - VL-BERT
 - UNITER
 - Oscar
 - 12-in-1
- 4. 最后

0. 本文结构

- VQA任务是什么
- 相关的数据集
- 介绍主流的模型和方法 (2014-2020)

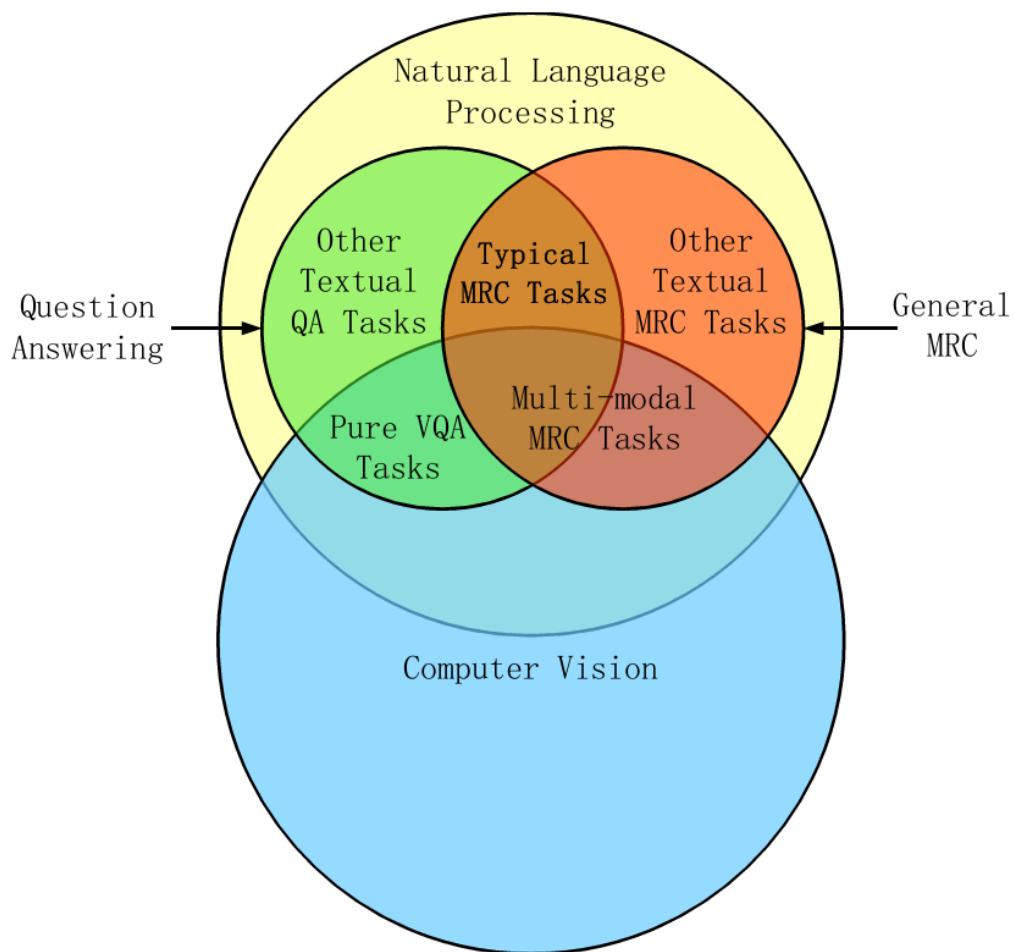
1. VQA任务简介

VQA 介于图像理解 (CV) 和自然语言处理 (NLP) 的交集。VQA 任务的目的是开发出一种系统来回答有关输入图像的特定问题。答案可以采用以下任何形式：单词，短语，二元答案，多项选择答案或文本填空。

在 CV 领域，CNN 是当前非常重要的基础模型。进而产生了VGGNet, Inception, ResNet等模型。类似的，NLP 领域，RNN 是之前主要的模型架构，因为LSTM的引入使得 RNN 有了重大突破。如 Vanilla VQA 模型使用了 VGGNet 和 LSTM 相结合的方法。后来在NLP领域的注意力机制（Attention Mechanism）也开始在CV领域开始得到应用。就有了Stacked Attention Network等。2018年 BERT 横空出世，在 NLP 领域掀起了革命。所以近两年，BERT 也开始进入到VQA任务中，BERT一开始是用于替换 RNN 来处理文本。但是在2019,2020年开始，一些模型（如，VL-BERT）开始把简单有效的Transformer模型作为主干并进行拓展，视觉和语言嵌入特征可以同时作为输入。然后进行预训练以兼容下游的所有视觉-语言联合任务。

VQA和NLP任务的区别

我们来看下面的这张图

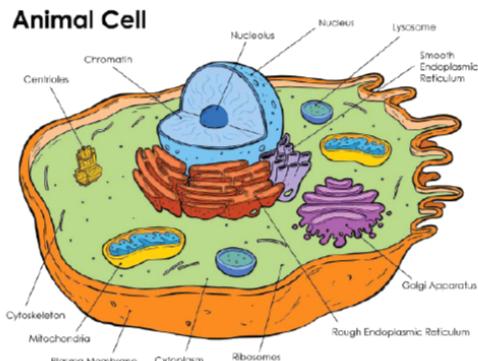


其中，machine reading comprehension (MRC)和question answering (QA)的关系其实是相对独立的。在本图中，Pure VQA任务一般是没有引入**额外的context**，只是单纯的有{图， 问句， 答案}。而Multi-

modal MRC任务，实际上就只是引入了额外的context作为VQA任务的知识，并且更加注重于自然语言的理解。下图可以给出一个来自TQA数据集的例子。（该数据集主要来自课本）

Passage with illustration:

This diagram shows the anatomy of an Animal cell. Animal Cells have an outer boundary known as the plasma membrane. The nucleus and the organelles of the cell are bound by this membrane. The cell organelles have a vast range of functions to perform like hormone and enzyme production to providing energy for the cells. They are of various sizes and have irregular shapes. Most of the cells size range between 1 and 100 micrometers and are visible only with help of microscope.

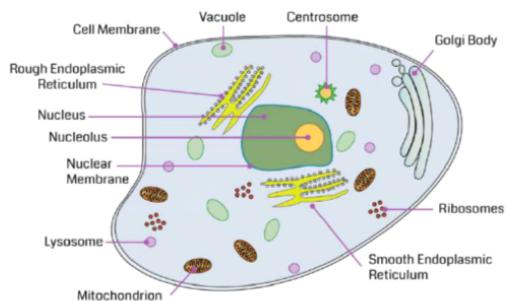


Question with illustration:

What is the outer surrounding part of the Nucleus?

Choices:

- (1) Nuclear Membrane ✓
- (2) Golgi Body
- (3) Cell Membrane
- (4) Nucleolus



既然讲到了MRC不妨提一下，MRC的主要任务类型一共有四种，分别为完形填空（Cloze Style）、多项选择（Multiple Choice）、片段抽取（Span Prediction）和自由作答（Free-form Answer）。大多数现有的MRC任务都是文本问题解答任务，因此将这种机器阅读理解任务视为典型的机器阅读理解任务（Typical MRC）。

关于VQA和Textual Question Answering (TQA) 的不同，主要是数据集信息形式的不同。

Visual Question Answering		Textual Question Answering	
Visual		Textual	Context: There is a cat and a dog in the image. A cat is on the table. A grey cat. A black dog. A dog is next to the fen. A plant is next to the cat. A green plant. The cat is looking at the plant.
Textual	Question1: What color is the cat? Question2: Which animal in this image is able to climb trees? (Extra Knowledge: Cat is able to climb trees.)	Textual	Question1: What color is the cat? Question2: Which animal in this image is able to climb trees? (Extra Knowledge: Cat is able to climb trees.)
Textual	Answer1: grey Answer2: cat	Textual	Answer1: grey Answer2: cat

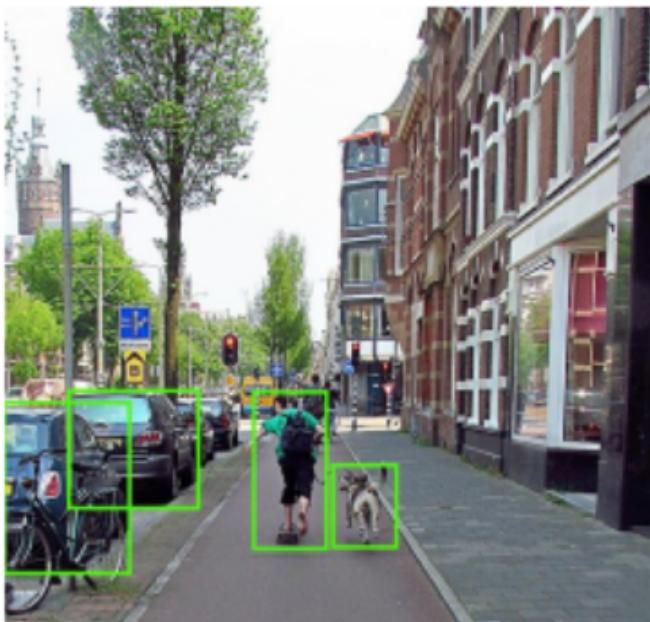
VQA和CV任务的区别

- VQA 的总体目标是从图像中提取与问题相关的语义信息，从细微物体的**检测**到抽象场景的**推理**。
- 大多数 CV 任务都需要从图像中提取信息，但与 VQA 相比都存在某些局限性。
- 但是实际上，由于 VQA 中问题会提供一定的场景，在这个场景下，答案的粒度是一定的。并且是有**明确的**答案，所以相对来说 VQA 的评价要相对简单一些。

基于对象检测的任务

- 对象识别、动作识别和场景分类都可以被定义为**图像分类任务**，现在最好的方法是使用 CNN 进行训练，将图像分类为特定的语义类别。
 - **对象识别**一般只需要对图像中的主要对象进行分类，而不用理解其在整个场景中的空间位置或作用。
 - **目标检测**通过对图像中每个对象实例放置一个边界框来定位特定的语义概念。
 - **语义分割**通过将每个**像素**分类为一个特定的语义类，使定位的任务更进一步。
 - **实例分割** (Instance segmentation) 用于区分同一语义类的不同实例。

标签歧义



左：目标检测，右：语义分割

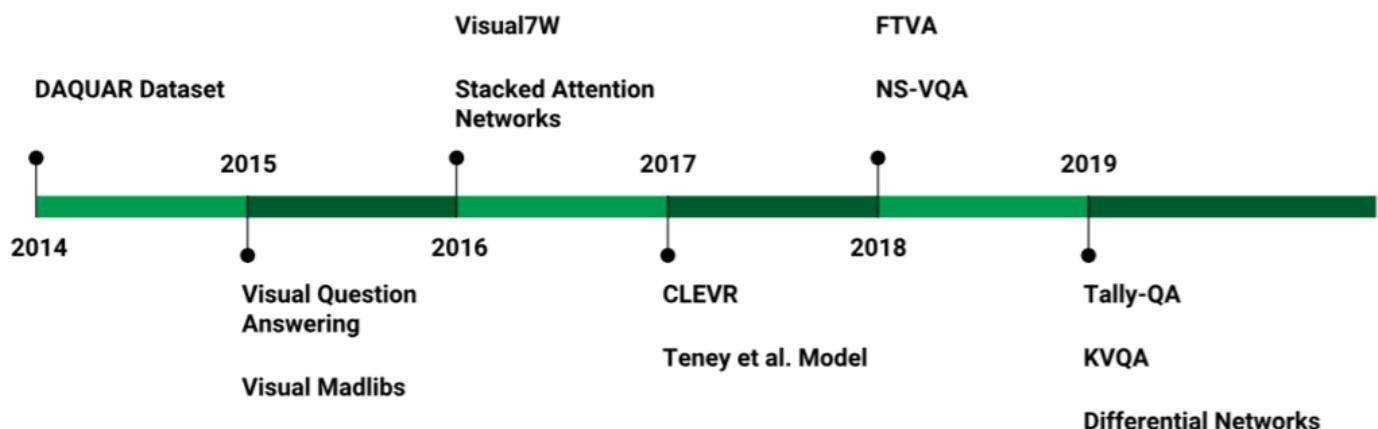
- **语义分割**或**实例分割**都不足以全面理解整个场景；
- 其中主要的问题在于**标签歧义** (label ambiguity)
 - 比如上述图中“**黄叉**”的位置取 "bag"、"black"、"person" 之一都没有问题。
 - 一般来说，具体选取哪个标签，取决于具体的任务。
- 此外，目前的主流方法 (CNN+标签) 不足以理解物体在整个场景下的作用 (role)
 - 比如，将“**黄叉**”位置标记为 "bag" 不足以了解该包与人的关系；或者标记为 "person" 也不能知道这个人的状态 (跑、坐、...)

- 理想的 VQA 要求能够回答关于图像的任意问题，因此除了基本的检测问题，还需要理解对象彼此，以及和整个场景之间的关系。

图像描述任务

- 除了 VQA 外，**图像描述** (image captioning) 是另一个比较主流的、需要结合 CV 和 NLP 的任务。图像描述任务的目标是对给定图像生成相关的自然语言描述。
- 结合 NLP 中的一些方法 (RNN等)，生成描述有不同的解决方案。
- 但是，图像描述的另一个难点是**评价**。
 - 一些自动评价方法：BLEU、ROUGE、METEOR、CIDEr
 - 这些方法中，除了 CIDEr，最初都是为了评价机器翻译的结果而提出的。
 - 这些方法每一个都存在一些局限性，它们常常将由机器生成的标题排在人工标题之前，但从人的角度看，这些结果并不够好，或者说**不是目标描述**。
- 评价的一个难点在于，给定图像可以存在许多有效的标题，这些标题可以比较宽泛，也可能很具体。
 - 比如上面的图中既可以描述为"A busy town sidewalk next to street parking and intersections."；
 - 也可以使用 "A woman jogging with a dog on a leash."
- 如果不加限制，图像描述系统总是倾向于生成**得分更高的**表述。
 - 比如 "A person is walking down a street" 或 "Several cars are parked on the side of the road" 这些普适的描述总是会得到较高的排名 (Rank) 。
 - 事实上，一个简单图像描述系统，只要使用 KNN 等方法找到与给定图像比较**相似的图像**，并把它们的描述返回就能在部分评估指标下得到不错的分数。

2. VQA 相关的数据集 (datasets)



- DAQUAR (Dataset for Question Answering on Real World Images)
- COCO-QA
- VQA Dataset (Visual Question Answering dataset)
- Visual Madlibs
- Visual Genome

- Visual7W
- CLEVR
- Tally-QA
- KVQA

DAQUAR

Dataset for Question Answering on Real World Images 真实世界图像问答数据集

该数据集很小，共有1,449张图像。题库包括12,468个问答对和2,483个独特问题。通过人工注释生成问题，并使用NYU-Depth数据集的注释将问题限制在9个问题模板中。包含了**6795张训练数据**和**5673张测试数据**，所有图像来自于数据集NYU-DepthV2 Dataset。该数据集质量较差，一些图像杂乱无章，分辨率低，并且问题和回答有明显的语法错误。

- 训练集：6795，测试集：5673

相关论文：A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input [6]

下载地址：[Visual Turing Challenge](#)



QA: (What is the object on the counter corner?, microwave)



QA: (How many doors are open?, 1)



QA: (Where is the oven?, on the right side of refrigerator)

该数据集只包含室内场景。关于在室外的问题就很难回答出来。对人类的评估，NYU数据集只有50.2%的准确率。

COCO-QA

COCO-QA数据集比DAQUAR大得多。它包含123,287张来自可可数据集的图片。



COCO-QA: What does an intersection show on one side and two double-decker buses and a third vehicle,?

Ground Truth: Building

- 训练集: 78,736, 测试集: 38,948
- 问题分布: **object** (69.84%), **color** (16.59%), **counting** (7.47%) and **location** (6.10%)
- 所有问题的答案都是一个单词, 只有435个独一无二的答案。 (可以做一个multiclass的任务)

缺点: QA pairs是用NLP算法生成的, 是将长句子划分成短句子处理的, 这就忽视了句子中的语法和从句问题, 算法结果不够智能。这导致一些 Q 存在语法错误甚至无法读懂。这个数据集另一个问题是, 它只有 4 种简单的问题 (因为这些问题基于图像的标题生成的)。这些问题可能只需要捕捉到图像中的一些**局部信息**就能得出答案。

| 相关论文: Exploring Models and Data for Image Question Answering [7]

| 下载地址: [COCO-QA Dataset](#)

VQA数据集

现在有两个版本, 分别为 VQA v1 (2015) 和 VQA v2 (2017)。

VQA v1

该数据集由两部分组成: COCO-VQA 和 SYNTH-VQA; 前者为真实图像, 后者为合成卡通图像。其中 SYNTH-VQA 由 50,000 个不同的模拟场景组成, 其中包含 100 多个不同的物体、30 个动物模型和 20 个人体模型。

VQA Dataset 为每幅图片提供了三个问题, 每个问题有十个答案。

Is something under the sink broken? yes yes yes	Can you park here? no no no	What kind of store is this? bakery bakery pastry	How many bikes are there? 2 2 3
What number do you see? 33 33 33	What color is the hydrant? white and orange white and orange white and orange	art supplies grocery grocery grocery	4 4 12
Is this man crying? no no	Has the pizza been baked? yes yes yes	Is the display case as full as it could be? no no yes	What number is the bus? 48 48 number 6
Does this man have children? yes yes yes	What kind of cheese is topped on this pizza? feta feta ricotta	How many pickles are on the plate? 1 1 1	
Is this man crying? no no	What is the shape of the plate? circle round round	What does the sign say? stop stop stop	What shape is this sign? octagon octagon octagon
How many glasses are on the table? 3 3 3	Do you think the boy on the ground has broken legs? yes yes yes	Are the kids in the room the grandchildren of the adults? probably yes yes	diamond octagon round
What is the woman reaching for? door handle glass wine fruit glass remote	Why is the boy on the right freaking out? his friend is hurt other boy fell down someone fell	What is on the bookshelf? nothing nothing nothing	How many balls are there? 2 2 1
2 2 6	ghost lightning sprayed by hose	books books books	right right left
			right side right side left side

Fig. 2: Examples of questions (black), (a subset of the) answers given when looking at the image (green), and answers given when not looking at the image (blue) for numerous representative examples of the dataset. See the appendix for more examples.

- COCO-VQA: 614,163 total, with 248,349 for training, 121,512 for validation, and 244,302 for testing
- SYNTH-VQA: 150,000 QA pairs

缺点：人工标注的开放式问答数据集，但是VQA v1有很多类语言偏见（bias）。自然图像数据集往往具有更一致的上下文和偏见，例如，街头场景比斑马更有可能有狗的图片。

相关论文：VQA: Visual Question Answering [8]

下载地址：[Visual Question Answering v1](#)

VQA v2

2017年更新为VQA v2.0，包含使用真实图片的VQA-real和卡通图片的VQA-abstract。VQA-real包含123287 training和81424 test images from COCO，由真人提供开放型和是非型问题和多种候选答案，共614163个questions。VQA-abstract包括50000scenes，每个scene对应3个questions。相较于VQAv1尽量减少了语言偏见（为每个问题补充了图片），但是仍存在一些偏见。



VQA-real [3]

Q: What shape is the bench seat ?
A: oval, semi circle, curved, curved, double curve, banana, curved, wavy, twisting, curved

Q: What color is the stripe on the train ?
A: white, white, white, white, white, white, white, white, white, white

Q: Where are the magazines in this picture ?
A: On stool, stool, on stool, on bar stool, on table, stool, on stool, on chair, on bar stool, stool



VQA-abstract [3]

Q: Who looks happier ?
A: old person, man, man, man, old man, man, man, man, man, grandpa

Q: Where are the flowers ?
A: near tree, tree, around tree, tree, by tree, around tree, around tree, grass, beneath tree, base of tree

Q: How many pillows ?
A: 1, 2, 2, 2, 2, 2, 2, 2, 2

相关论文：Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering [9]

下载地址：[Visual Question Answering v12](#)

Visual Madlibs

在2015年发布。图像是从MS-COCO收集的。使用模板和对象信息自动生成描述性的填空题。由3名AMT工人组成的小组回答以此方式产生的每个问题。答案可以是单词或短语。空白的多种选择也作为附加评估基准提供。数据集包含10,738张图像和360,001个问题。在准确性度量标准上评估多项选择题。这一个填空的数据集。



This place is a(n) road.
When I look at this picture, I feel free.

The most interesting aspect of this picture is the motorcycles.
One or two seconds before this picture was taken, they stopped to chat and decided where to go.
One or two seconds after this picture was taken, the bikers ride down the road.



This place is a(n) restaurant.
When I look at this picture, I feel like I want donuts.

The most interesting aspect of this picture is the box of donuts.
One or two seconds before this picture was taken, the box was closed.
One or two seconds after this picture was taken, a third person out of frame picks up a donut.



This place is a(n) waterway.
When I look at this picture, I feel concerned.
The most interesting aspect of this picture is the men standing on elephants.
One or two seconds before this picture was taken, the people were sitting on the elephant.
One or two seconds after this picture was taken, the men got off the elephants.



Person A is a girl.
Person B is learning how to surf.
Person A is kneeling on a surfboard.
Person B is a man in blue shorts.
Person B is walking on a beach.
Person B is at the beach.

Person C is a short haired black girl.
Person C is practicing surfing.
Person C is on a gold surfboard.
Person D is a lady in board shorts.
Person D is standing around.
Person D is next to a blue surfboard.



Person A is wearing a dark suit.
Person A is looking at an elephant.
Person A is at a zoo.
Person B is wearing a grey T-shirt.
Person B is talking to two other people.
Person B is next to an elephant.



Person A is a balding male.
Person A is playing a video game.
Person A is downstairs.
Person B is wearing jeans.
Person B is playing wii.
Person B is in a basement.



Person A is a young man in green.
Person A is trying to block a frisbee.
Person A is on a field.
Person B is wearing purple.
Person B is throwing a frisbee.
Person B is on a field.



The couches are white.
The couches are in the center of the room.
The TV is on.
The TV is near the wall.
People could relax on the couches.



The car is white.
The car is on a concrete pad.
The umbrellas are open.
The car is on a concrete pad.
The umbrellas are in the people's hands.
People could ride in the car.
People could stay dry under the umbrellas.



The person is putting food in the bowl.



The people are eating cake at the dining table.
The people are serving the cake.

Figure 2: Madlibs description. The first row corresponds to question types 1-5, the second row corresponds to question types 9-11, and the third row is to question types 6-8 and question type 12. All question types are listed in Table 1.

相关论文：Visual Madlibs: Fill in the blank Image Generation and Question Answering [10]

下载地址：[Visual Madlibs](#)

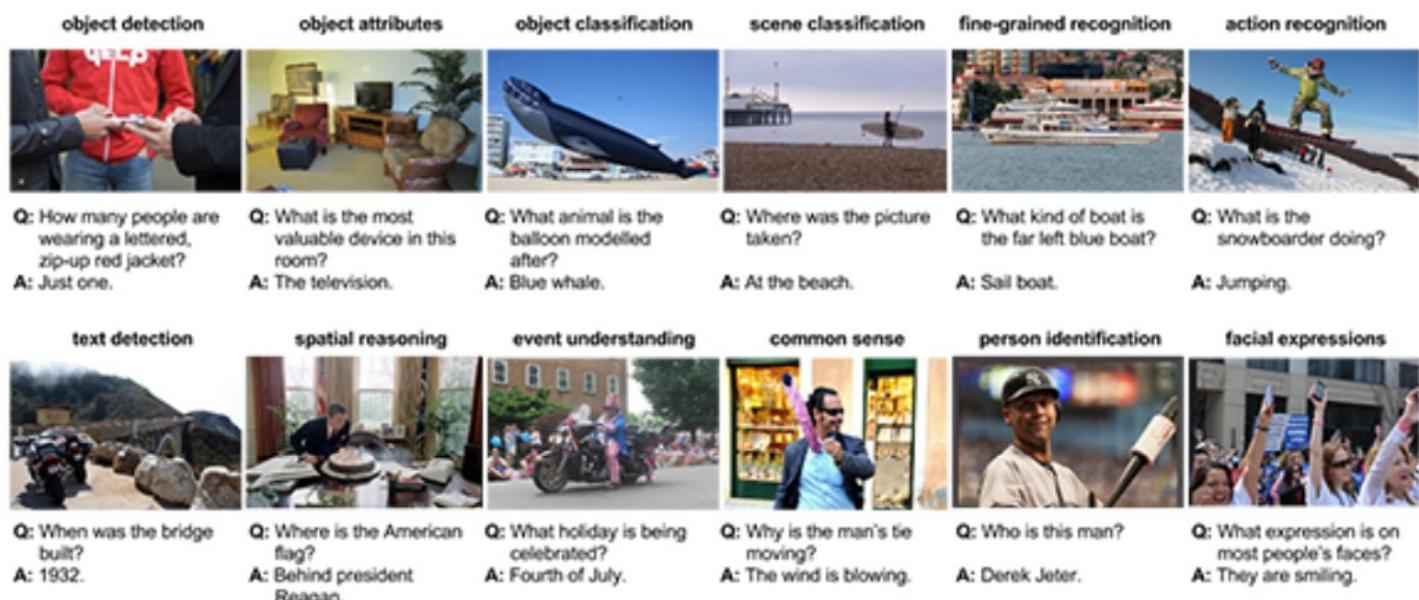
Visual Genome

在2017年发布。包含图像108,249张和170万个QA Pairs，每个图像平均17个QA对。图像来源是YFCC100M和COCO数据集，共有约540万张图像中的区域描述信息，这些信息能够达到精细的语义层次，问题类型是6W (what, where, how, when, who, why)，数据集并没有对训练和测试数据进行切分。

QA的收集有两种方法，一种是 free-form method 随意人为提问（会出现相似问题或对图像全局内容提问），另一种是 regionspecific method 针对图像中的特定区域提问。

该数据集中没有二值类问题。

答案多样性。Visual Genome 中最常见的 1000 个答案仅占数据集中所有答案的 65%，而 COCO-VQA 占82%，DAQUAR COCO-QA 占100%。 Visual Genome 只有 57% 的答案是单个词；而 COCO-VQA 中为 88%，COCO-QA 为100%，DAQUAR 为 90%。



相关论文：Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations [11]

下载地址：[The Visual Genome Dataset](#)

Visual7W

Visual7W 是 Visual Genome 的一个子集。7W 指的是 "What, Where, How, When, Who, Why, Which"。

Visual7W 也基于MS-COCO数据集。它包含47,300张COCO图像和327,939对QA pairs。该数据集还包含1,311,756个选择题和561,459个基础的答案。它主要由两种类型的问题组成。讲的问题是基于文本的，给出了一种描述。指向性问题是以"Which"开头的，必须由一组合理答案中的边界框正确识别。

回答是多选式的，每一个问题有4个候选答案，其中只有一个正确的。数据集不包含二值问题。

两类问题

- "telling" questions：答案是基于文本的
- "pointing" questions：以 Which 开头的问题，对于这些问题，算法必须在备选方案中选择正确的边界框。

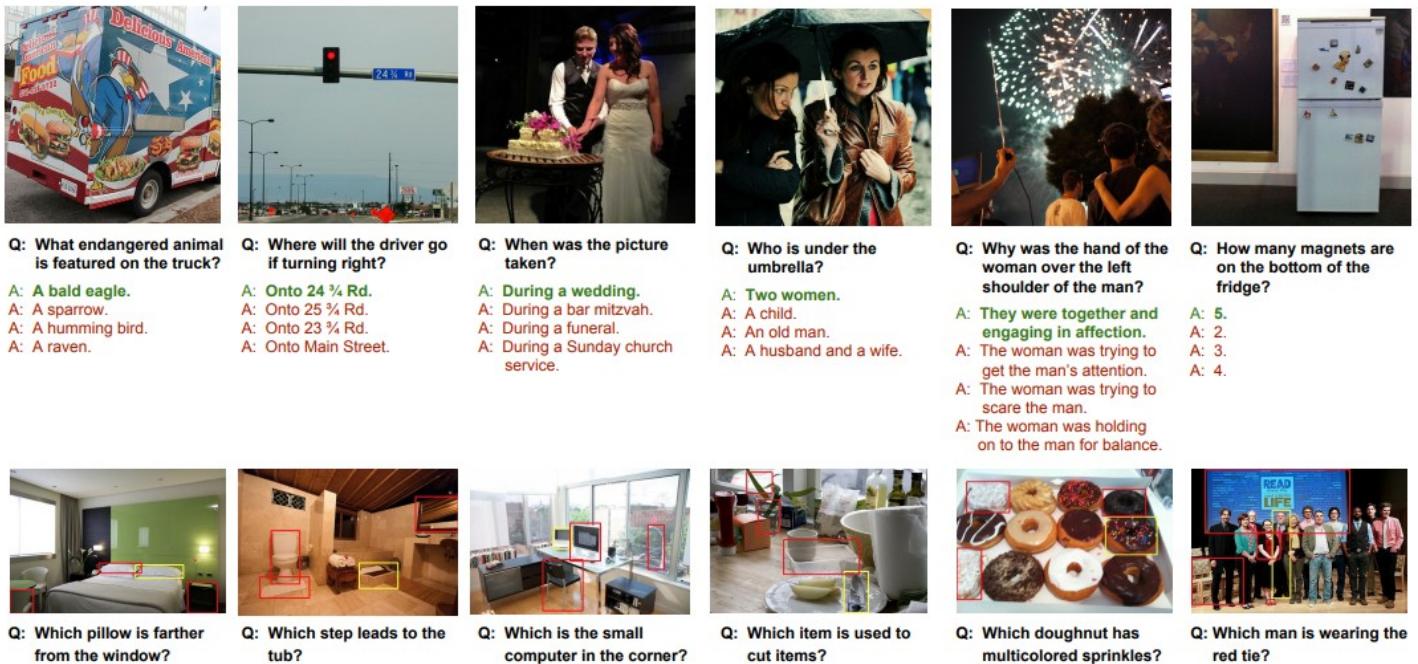


Figure 2: Examples of multiple-choice QA from the 7W question categories. The first row shows *telling* questions where the green answer is the ground-truth, and the red ones are human-generated wrong answers. The *what*, *who* and *how* questions often pertain to recognition tasks with spatial reasoning. The *where*, *when* and *why* questions usually involve high-level common sense reasoning. The second row depicts *pointing (which)* questions where the yellow box is the correct answer and the red boxes are human-generated wrong answers. These four answers form a multiple-choice test for each question.

相关论文：Visual7W: Grounded Question Answering in Images [12]

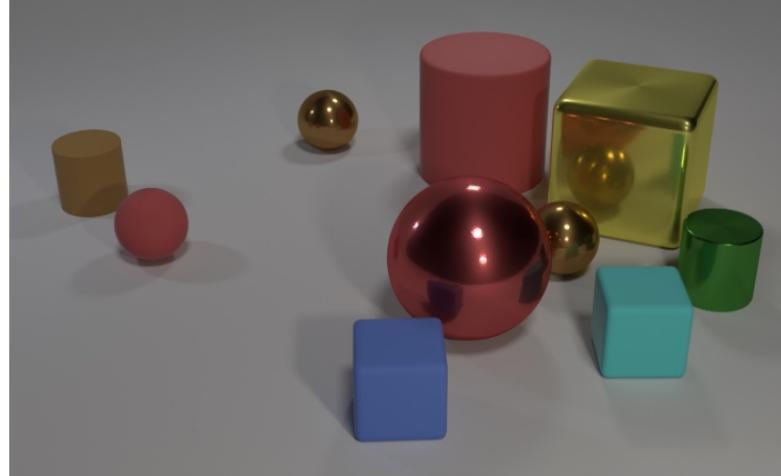
Github: <https://github.com/yukezhu/visual7w-toolkit>

CLEVR

CLEVR 是一个综合数据集，用于测试对VQA系统的视觉理解。

视觉推理在这里主要是限定在VQA(Visual Question Answering)的问题上，也就是让计算机看一副图，然后给出一个问题，让其回答。相比传统的VQA问题，视觉推理问题的要求是要让问题难度提升，必须经过**推理**才能回答。

该数据集是在每个图像中使用三个对象生成的，即圆柱体，球体和立方体。这些对象具有两种不同的大小，两种不同的材料，并以八种不同的颜色放置。



Q: Are there an **equal number** of **large things** and **metal spheres**?

Q: **What size** is the **cylinder** that is **left of** the **brown metal** thing that **is left of** the **big sphere**?

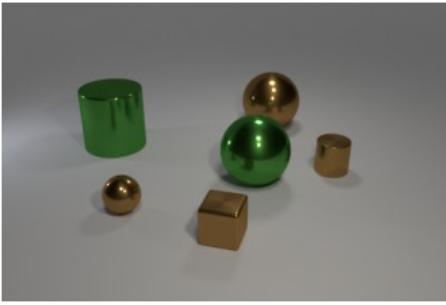
Q: There is a **sphere** with the **same size as** the **metal cube**; is it **made of the same material as** the **small red sphere**?

Q: **How many** objects are **either small cylinders or red** things?

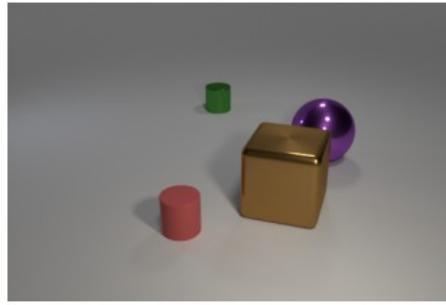
从上图可以看到，CLEVR数据集的图都是一些简单的几何体，但是问题却复杂的多。比如说上图的第一个问题：大物体和金属球的数量是一样的吗？为了能回答这个问题，我们首先需要找出大的物体还有金属球，然后要分别计算各自的数量，最后判断两者的数量是不是相等，也就是为了回答这么一个问题，我们需要三步的推理。

所有问题分为了5类：属性查询 (querying attribute) , 属性比较 (comparing attributes) , 存在性 (existence) , 计数 (counting) , 整数比较 (integer comparison) 。所有的问题都是程序生成的。该数据集的人为标注数据子集为CLEVR-Humans.

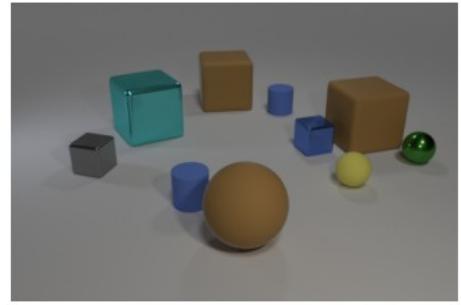
- A training set of 70,000 images and 699,989 questions
- A validation set of 15,000 images and 149,991 questions
- A test set of 15,000 images and 14,988 questions



- Q:** What color is the small shiny cube?
A: brown
Q-type: query_color
Size: 6
- Q:** There is a tiny shiny sphere left of the cylinder in front of the large cylinder; what color is it?
A: brown
Q-type: query_color
Size: 13



- Q:** What number of metal objects or purple matte things?
A: 0
Q-type: count
Size: 9
- Q:** Does the large cylinders are purple purple shiny object have the same shape as the tiny object that is behind the matte thing?
A: no
Q-type: equal_shape
Size: 14



- Q:** There is a large brown block in front shiny object; are there of the tiny rubber any blue shiny cubes cylinder that is behind behind it?
A: no
Q-type: exist
Size: 9
- Q:** There is a big cyan block; are there any big cyan metallic cubes that are to the left of it?
A: yes
Q-type: exist
Size: 20

- Q:** Is there a tiny thing that has the same material of the tiny material as the brown cylinder?
A: yes
Q-type: exist
Size: 7
- Q:** What is the object to the right of the brown shiny ball behind the tiny shiny cylinder?
A: metal
Q-type: query_material
Size: 14

- Q:** There is a object that is both on the purple shiny object; left side of the brown metal block and in front of the large purple shiny ball; how big is it?
A: small
Q-type: query_size
Size: 16
- Q:** There is a big what shape is it?
A: sphere
Q-type: query_shape
Size: 6

- Q:** The cyan block that cylinders have the same color as the tiny gray thing is what size?
A: large
Q-type: query_size
Size: 9

相关论文：CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning [13]

下载地址：[CLEVR](#)

Tally-QA

最近，在2019年，提出了Tally-QA数据集，它是开放式任务中最大的对象计数数据集。数据集包括简单和复杂的问题类型。该数据集的数量非常大，是VQA数据集的2.5倍。数据集包含287,907个问题，165,000个图像和19,000个复杂问题。



(a) How many giraffes are there?
GT: 2, DETECT: 2, Zhang: 2, RCN: 2



(b) How many people are standing?
GT: 2, DETECT: 4, Zhang: 3, RCN: 2



(c) How many people in the front row?
GT: 8, DETECT: 22, Zhang: 6, RCN: 8



(d) How many chairs have a girl sitting on them?
GT: 1, DETECT: 7, Zhang: 2, RCN: 1



(e) How many players are wearing red uniforms?
GT: 3, DETECT: 11, Zhang: 4, RCN: 3



(f) How many strings does the instrument to the left have?
GT: 4, DETECT: 3, Zhang: 1, RCN: 0

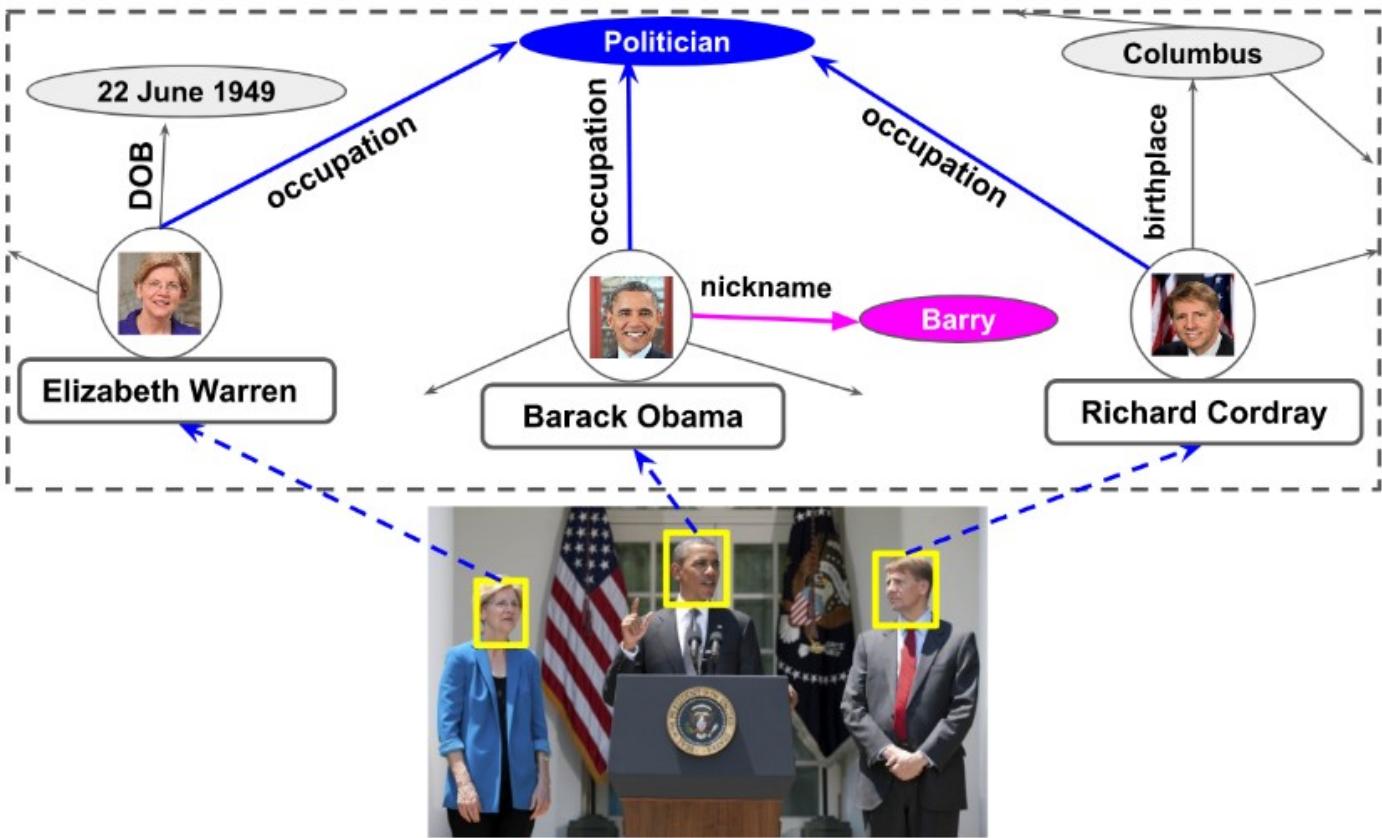
相关论文: TallyQA: Answering Complex Counting Questions [14]

Github: <https://github.com/manoja328/tallyqacode>

主页链接: [Tally-QA](#)

KVQA

最近对常识性问题的兴趣导致了基于世界知识的VQA数据集的发展。数据集包含针对各种名词的问题，并且还需要世界知识才能得出解决方案。此数据集中的问题需要对大型知识图（KG）进行多实体，多关系和多跳推理才能得出答案。数据集包含24,000张图像，包含183,100个问答对，使用约18K个专有名词。



Q: Who is to the left of Barack Obama?

A: Richard Cordray

Q: Do all the people in the image have a common occupation?

A: Yes

Q: Who among the people in the image is called by the nickname Barry?

A: Person in the center

相关论文：KVQA: Knowledge-aware Visual Question Answering^[15]

主页链接：[KVQA](#)

3. 主流模型与方法

一般来说，我们可以在VQA中概述这些方法：

- 从问题中提取特征。 (LSTM, GRU, BERT)
- 从图像中提取特征。 (VGGNet, ResNet, GoogLeNet, ImageNet)
- 结合这些特征来生成一个答案。 (目前主要有基于**分类**和**生成**两种方法)

3.1 非深度学习方法

回答类型预测 Answer Type Prediction (ATP)

来自论文：Answer-Type Prediction for Visual Question Answering^[16]

(Kafle and Kanan, 2016) 提出了VQA的贝叶斯框架，其中他们预测问题的答案类型并使用它来生成答案。可能的答案类型因其考虑的数据集而异。例如，对于COCO-QA，他们考虑四种答案类型：对象，颜色，计数和位置。

- 他们的模型根据图像x和问题q计算出答案a和答案类型t的概率。
- 使用语义分割来识别图像中的对象及其位置。
- 他们使用ResNet来处理图像，并跳级思考向量 (skip-thought vectors) 来处理文本。
- 然后，利用贝叶斯算法对目标的空间关系进行建模，计算出每个答案的概率。
- 是较早的 VQA 解决方案，但其有效性不如简单的基线模型；部分原因在于其依赖语义分割的结果。

多元世界问答 Multi-World QA

来自论文：A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input [17]

(Malinowski and Fritz, 2014) 这篇论文将基于问题和图像的答案概率建模为

$$P(A = a|Q, W) = \sum_T P(A = a|T, W)P(T|Q)$$

这里 T 为隐藏变量，它对应于从问题语义分析器 (semantic parser) 得到的语义树 (semantic tree)。W 是世界，代表图像。它可以是原始图像或从分割块获得的附加特征。使用确定性评价 (deterministic evaluation) 函数来评估 $P(A|T, W)$ 。使用简单的对数线性模型得到 $P(T|Q)$ 。这个模型被称为 SWQA。

作者进一步将其扩展到多元世界的场景，用来模拟分割和分类标签的不确定性。不同的标签代表不同的 W，所以概率模型为

$$P(A = a|Q, W) = \sum_W \sum_T P(A = a|T, W)P(W|S)P(T|Q)$$

这里，S 是带有类标签分布的一组分割图像集。因此，从分布中抽样分割图像时将得到其对应的一个可能的 W。由于上述方程很复杂，作者仅从 S 中抽样固定数量的 W。

3.2 无注意机制的深度学习模型 Non-attention Deep Learning Models

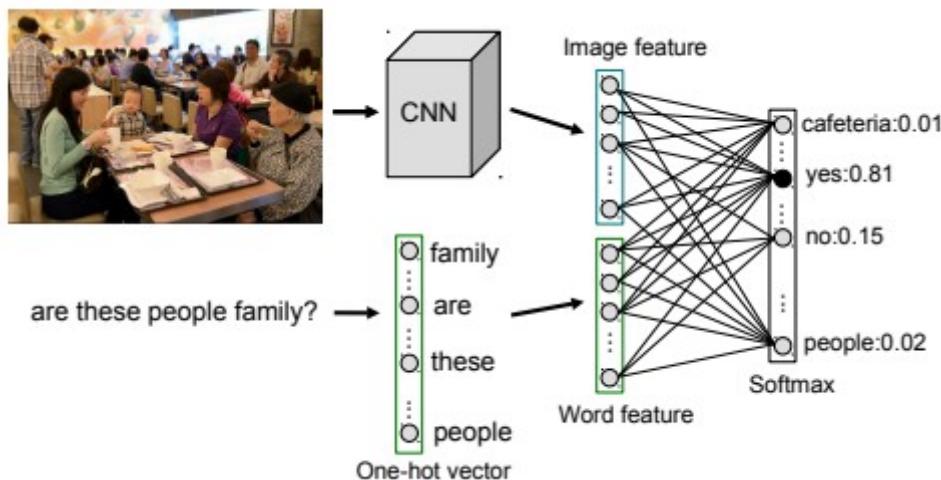
VQA 的深度学习模型通常使用卷积神经网络 (CNN) 来嵌入图像与循环神经网络 (RNN) 的词嵌入 (word embedding) 来嵌入问题。这些嵌入以各种方式组合和处理以获得答案。

iBOWING

- CNN (GoogLeNet)
- BoW

相关论文：Simple Baseline for Visual Question Answering [18]

(Zhou, 2015) 提出了一种叫做 iBOWING 的基线模型。他们使用预训练的 GoogLeNet 图像分类模型的层输出来提取图像特征。问题中每个词的词嵌入都被视为文本特征，因此文本特征是简单的词袋 (bag-of-word)。连接图像和文本的特征，同时对答案分类使用 softmax 回归。结果表明，该模型在 VQA 数据集上表现的性能与几种 RNN 方法相当。



作者的灵感来源于早期的一篇文章，BOWIMG baseline (Bag-of-words + image feature) 在COCO数据集上的效果要比LSTM要好一些，但是在更大一些的COCO VQA数据集上，BOWIMG baseline却表现比LSTM更糟。基于此，作者提出了iBOWIMG模型。

训练过程中作者提到了两个小细节：

- Learning rate and weight clip (学习率和权值截取)：作者发现设置不同的学习率和权值截取对于词嵌入和softmax都有性能的提升。在词嵌入层的学习率要高于softmax的学习率。
- Model parameters to tune (模型参数微调)：需要调整的有3个参数，训练的epoch，权值截取和学习率，低频QA的阈值。

Github: <https://github.com/metalbubble/VQAbaseline>

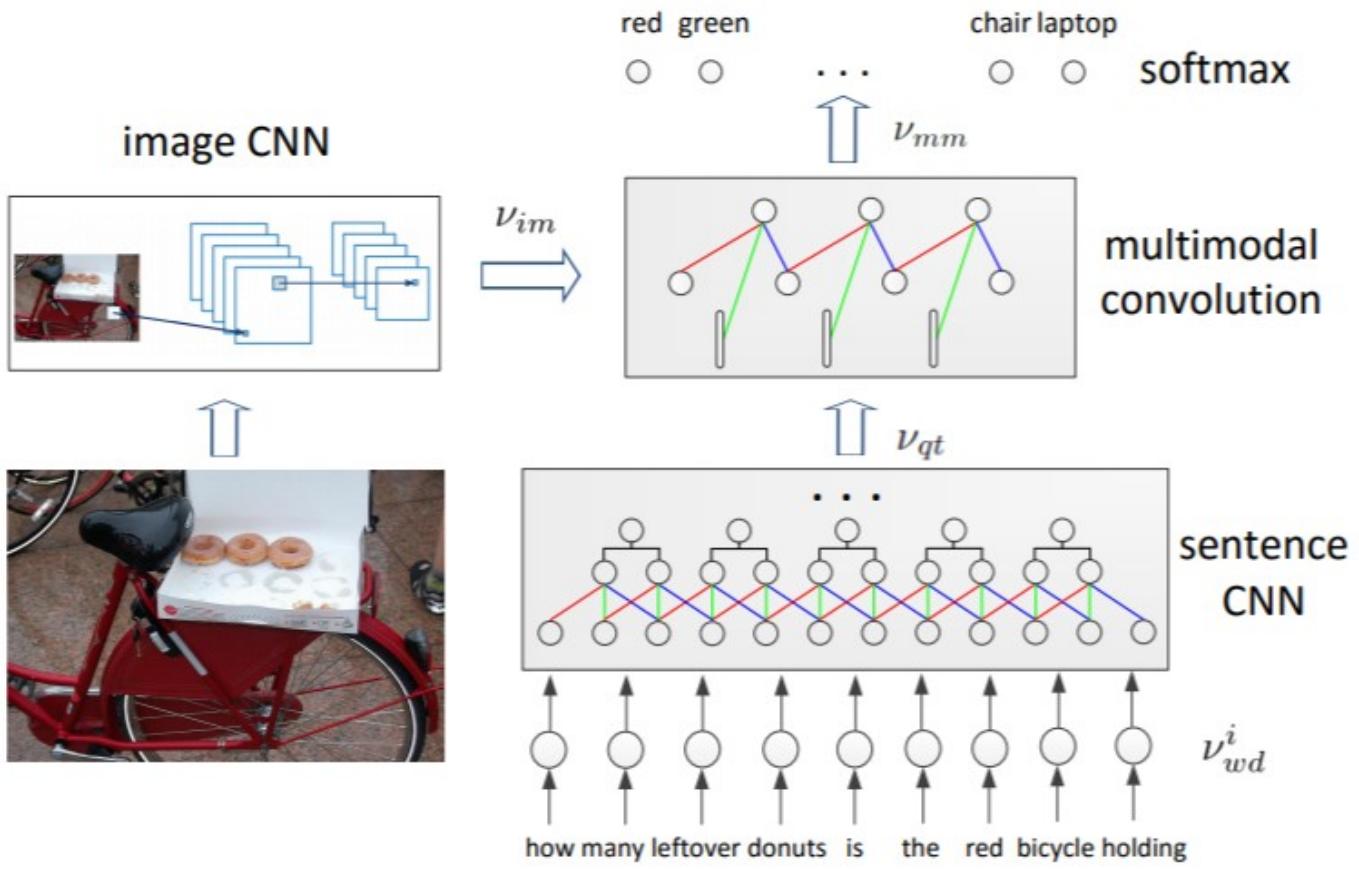
Full-CNN

- CNN only

相关论文：Learning to Answer Questions From Image Using Convolutional Neural Network [19]

(Ma, 2015) 提出了一种仅用 CNN 的模型，称为 Full-CNN。模型使用三种不同的 CNN。

- 编码图像
- 编码问题
- 将图像和问题的编码结合在一起并产生联合表征



图像 CNN 使用与 VGG 网络相同的架构，并从该网络的第二层获取长度为 4096 的向量。这通过另一个完全连接的层，以获得大小为 400 的图像表征向量。

句子 CNN 涉及 3 层卷积和最大池化 (max pooling)。卷积感受野 (receptive field) 的大小设置为 3。换句话说，核函数 (kernel) 会计算该词及其相邻的邻居。

联合 CNN 称为多元模态 CNN (multi-modal CNN)，在问题表征上的卷积感受野大小为 2。每个卷积运算都在完整的图像上进行。将多元模态 CNN 的最终表征结果传入 softmax 层以预测答案。

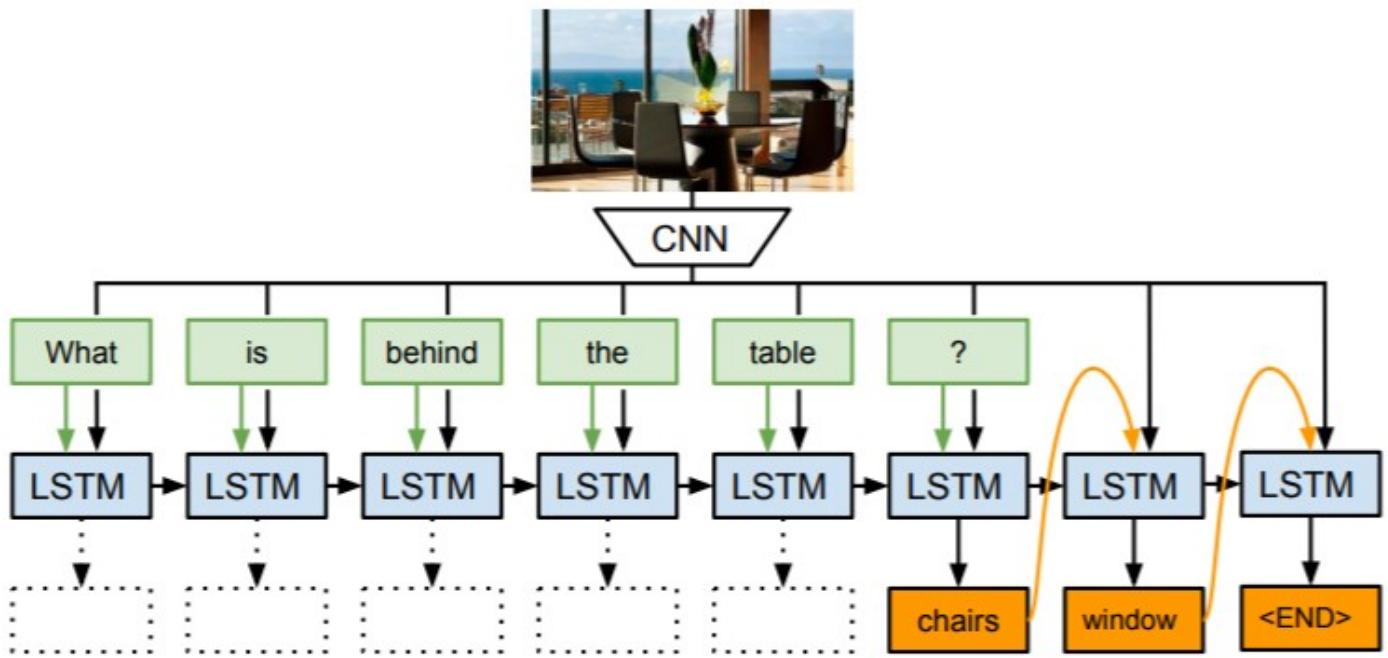
神经元询问 Ask Your Neurons (AYN)

- CNN
- RNN (LSTM)

相关论文：Ask Your Neurons: A Deep Learning Approach to Visual Question Answering [20]

(Malinowski, 2016) 以CNN和LSTM为基础，以一种新的使用方式，设计了一个预测结果长度可变的模型。该模型将视觉问答任务视为结合图像信息作为辅助的sequence to sequence任务。

首先由一个预训练好的深度CNN模型抽出要回答的**图片特征**，然后将图片特征和转化为**词向量**的问题词一起送入LSTM网络，在每次送入一个问题词的**同时**将图片特征送入网络，直到所有的问题特征信息抽取完毕。接下来用同一个LSTM网络产生答案，直至产生结束符(\$)为止。该模型的训练过程是结合图像特征的LSTM网络的训练以及词向量的生成器的训练。



解码答案可以用两种不同的方式，一种是对不同答案的分类，另一种是答案的生成。分类由完全连接层生成输出并传入覆盖所有可能答案的 softmax 函数。另一方面，生成由解码器 LSTM 执行。在每个时间点的 LSTM 将前面生成的词以及问题和图像编码作为输入。下一个词使用覆盖词汇表的 softmax 函数来预测。需要注意的一点是，该模型在编码器和解码器 LSTM 之间共享一些权重。

Github: https://github.com/mateuszmalinowski/visual_turing_test-tutorial

Vis + LSTM

- CNN (VGG Net)
- RNN (LSTM)

相关论文：Exploring Models and Data for Image Question Answering [21]

论文 (Ren et al., 2015)有以下几点贡献：

- 提出一个end-to-end QA模型，这个模型利用visual semantic embedding 连接CNN, RNN。
- 提出一个自动问题生成算法，这个算法可以将描述图像的句子转化为问题
- 基于以上算法生成COCO-QA数据集

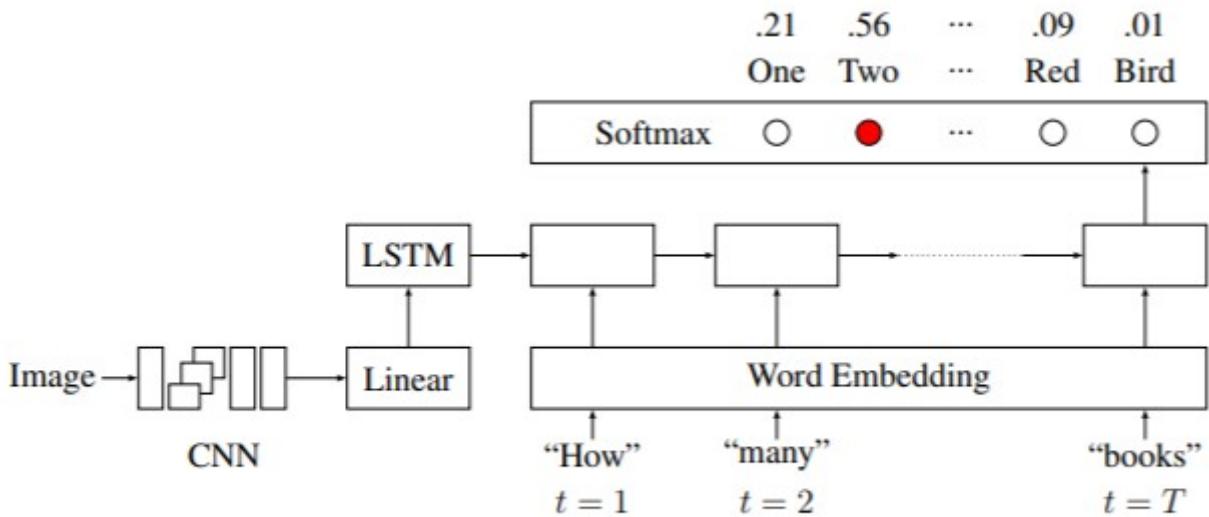


Figure 2: VIS+LSTM Model

该模型与 AYN 模型非常相似。该模型使用 VGG Net 的最后一层隐藏层作为visual embeddings，并且在训练期间保持CNN不变。与之前的模型相反，在编码问题之前，它们将图像编码作为第一个“词”传入 LSTM 网络。该 LSTM 的输出先通过完全连接层，然后通过 softmax 层。

作者还提出了一种使用双向 LSTM 的 2Vis+BLSTM 模型。向后的 LSTM 也将图像编码作为第一个输入。两个 LSTM 的输出相连接，然后通过一个 dense 和 softmax 层。

一共四个模型：

- Vis + LSTM
- 2-Vis + BiLSTM
- IMG + BOW
- FULL (以上三个模型的平均)

GitHub 1: https://github.com/abhshkdz/neural-vqa?utm_source=tuicool&utm_medium=referral

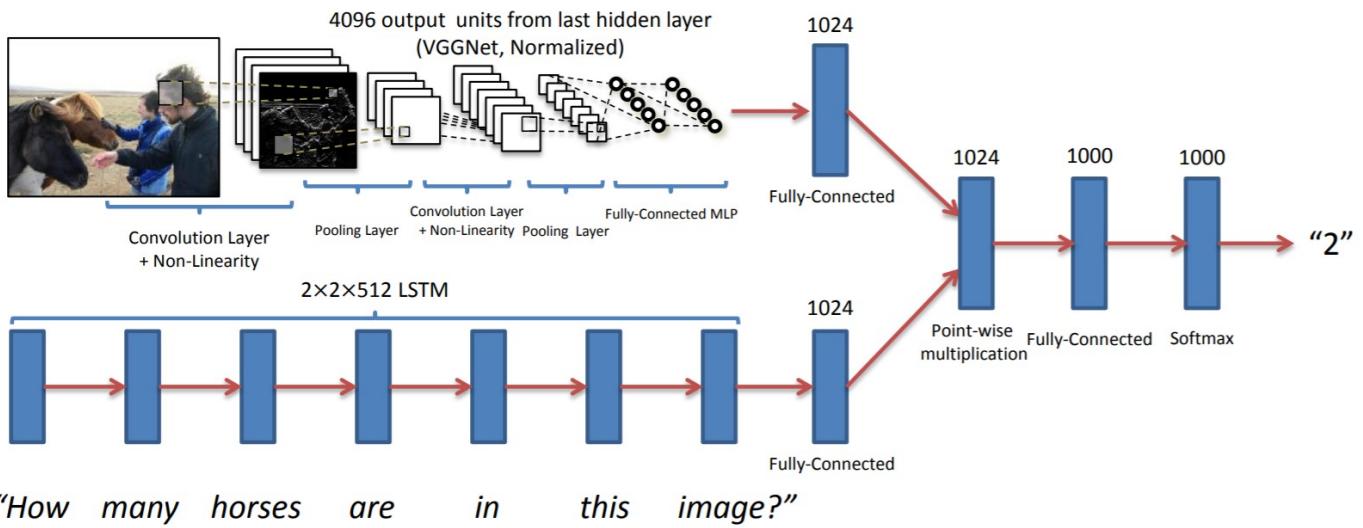
GitHub 2: <https://github.com/renmengye/imageqa-public>

GitHub 3: <https://github.com/VedantYadav/VQA>

Vanilla VQA (deeper LSTM Q + norm I)

- CNN (VGG Net)
- RNN (2 layer LSTM)

说老实话，这个模型的名字只有在"Visual Question Answering using Deep Learning: A Survey and Performance Analysis" [1:1] 这里看到过。实际上应该是在论文"VQA: Visual Question Answering" [8:1] 中所描述的 "deeper LSTM Q + norm I"。



图像：

- I: 利用 VGGNet最后一层隐藏层的激活作为 4096- dim 图像嵌入。
- norm I: 这些是在 VGGNet的最后一个隐藏层使用L2正则化激活。

问题：

- 词袋问题(BoW Q) 问题和答案的第一个单词有很强的相关性。选择前30个创建一个词袋。
- LSTM Q 具有一个隐藏层的lstm对1024维的问题进行嵌入。对每一个问题字进行编码，采用全连通层 + tanh 将其进行300维嵌入，然后供给LSTM。
- deeper LSTM Q: 使用具有两层隐藏LSTM将问题进行2048维嵌入，然后利用全连通层 + tanh 非线性函数将 2048-dim 嵌入变换为 1024维。

多层感知机：将图像和问题结合。首先通过全连通层+ tanh 非线性将图像嵌入变换为 1024-dim 来匹配问题的 LSTM 嵌入。转换后的图像和 LSTM 嵌入(在公共空间中)然后通过元素的乘法进行融合。

Github: https://github.com/GT-Vision-Lab/VQA_LSTM_CNN

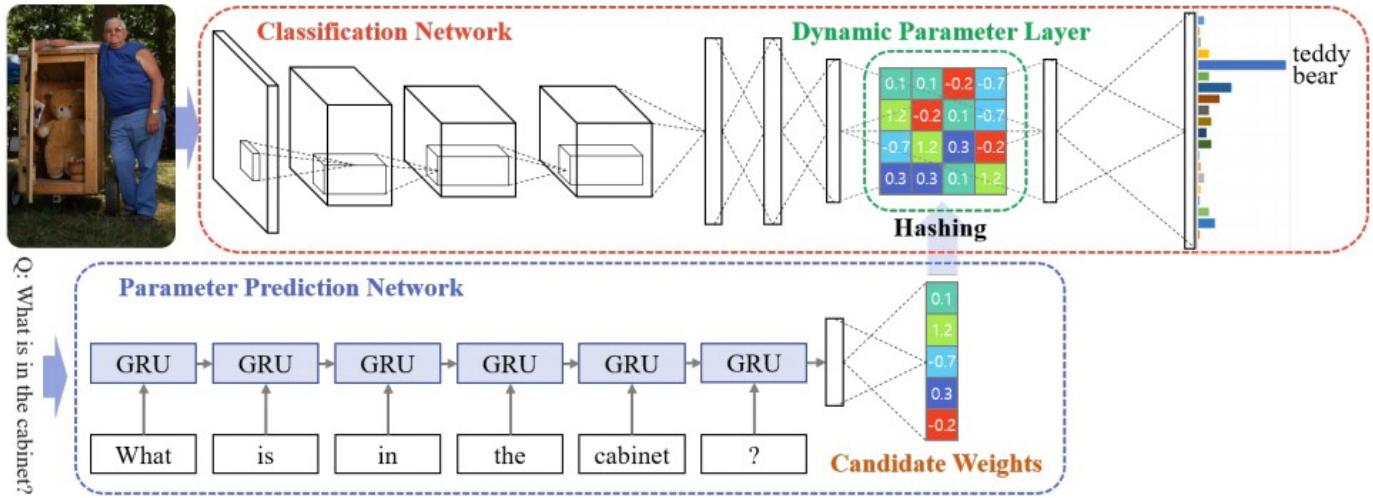
动态参数预测 Dynamic Parameter Prediction (DPPnet)

- CNN (VGG-16 -> 3 fully-connect + DPN)
- RNN (GRU)

来自论文：Image Question Answering using Convolutional Neural Network with Dynamic Parameter Prediction [22]

(Noh, 2016) 论文中的主要贡献：

- 采用CNNc+DPN处理ImageQA任务，DPN的参数根据给定问题动态生成
- 采用一个Hash trick对参数降维
- 通过在一个大的文本集上fine-tune GRU，提升网路的泛化性能
- 首次同时在DAQUAR,COCO-QA,VQA上进行实验



作者认为，设定一组固定参数并不足以满足 VQA 任务。他们采用 VGG-16 网络架构，删除最终 softmax 层，并添加三个全连接层，并最后使用覆盖所有可能答案的 softmax 函数。这些完全连接层的第 2 层没有固定的一组参数。

相反，**参数来自 GRU 网络**。该 GRU 网络用于对问题进行编码，并且 GRU 网络的输出通过完全连接层给出候选参数的权重小向量。然后使用逆哈希函数 (inverse hashing function) 将该向量映射到第 2 完全连接层所需的参数权重大向量中。这种哈希 (hashing) 技术被用于避免预测全部的参数权重而带来的计算成本高昂，并避免由此导致的过拟合。或者可以将动态参数层视为将图像表征和问题表征相乘得到的联合表征，而不是传统的以线性方式组合。

项目主页：<http://cvlab.postech.ac.kr/research/dppnet/>

Github: <https://github.com/HyeonwooNoh/DPPnet>

3.3 基于 Attention 的模型 Attention Based Models

对于 VQA 任务，注意机制模型聚焦在图像、问题或两者的重要部分，从而有效地给出答案。

例如，如果问题是“球是什么颜色的？”那么需要更加集中球所包含的图像区域。同样，在问题中，需要集中“颜色”和“球”这两个词，因为它们比其他的词更具信息性。

VQA 中，使用基于空间的 Attention 机制来创建**特定区域**的 CNN 特征，而不像基线模型中那样直接使用全局特征。

Attention 背后的基本思想是，图像中的某些视觉区域和问题中的某些单词对于回答给定的问题比其他区域或单词更能提供更多的信息。

Where To Look (WTL)

- 基于 Edge Boxes 的方法 (木匾检测)

相关论文：Edge Boxes: Locating Object Proposals from Edges [23]

- word2vec

主要的想法：学习语言和视觉区域的非线性映射将特征纳入 共同的**潜在空间**以确定相关性。

相关论文：Where to look: Focus regions for visual question answering [24]

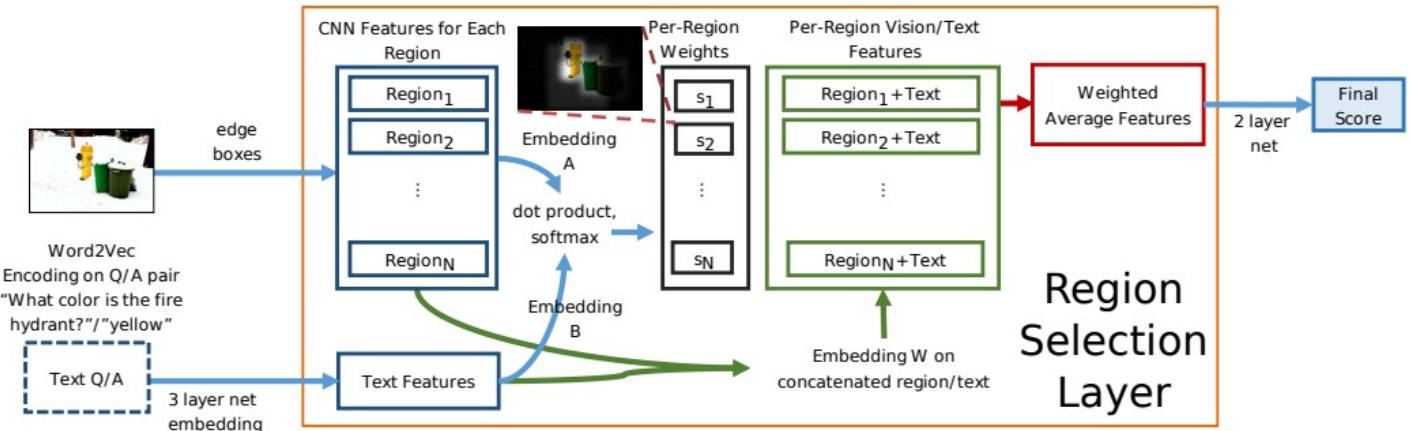


Figure 3. Overview of our network for the example question-answer pairing: “What color is the fire hydrant? Yellow.” Question and answer representations are concatenated, fed through the network, then combined with selectively weighted image region features to produce a score.

概括：where to look (Shih, 2016) 的地位有点相当于VQA 方向attention的始祖 第一次提出了基于QA的图像region attention 因为文章为2015的比较早 因此使用方法还存在不少瑕疵：具体做法为该网络只适用于mc类型的VQA 输入QA对，并置提取特征。图像过边缘检测得100分区，过cnn得特征、将每个region的向量与QA特征向量作内积得attention系数权值。最后与文本特征并置加权求和得weighted average features。然后过两个层得score，训练时的loss (hinge loss)。

整体模型：

- 图像先经过区域选择，对问题进行embedding操作；
- 用问题embedding对图像区域计算注意力权重；
- 融合问题特征和图像特征；
- 预测输出。

Step1: Image features :

通过edge boxes (边缘检测) 预训练网络得到 top99 region，然后全图算第100个region (注意：其中联合重叠阈值设定决定了区域的大小) 本task region 稍微小点好。作者猜测增加region number可能能够提升性能。用的VGG，取的最后一个隐藏层4096d和前一个softmax层1000d并置共5096d 因为1000那个包含物体类别信息。

Step2: Language representation :

首先将每个word通过Google News dataset进行预训练的w2v得到单词representation (相同词有相近的向量特征是open-ended前提) 之后通过4个Bin得到四种question sentence representation (而不是LSTM) 。

- Bin1：问题前两个词特征的平均
- Bin2：主语名词特征

- Bin3: 其他所有名词特征的平均
- Bin4: 去掉限定词和冠词之后的剩余词特征的平均
- Bin1+Bin2+Bin3+Bin4+answer representation = 1500维 这就是整个的representation

Step3: Image特征和QA特征都FC降维到900 然后点积后softmax成region probability

Step4: 最后的向量z过一个两层的fc后输出一个score 然后利用Hingeloss返回梯度

思考：

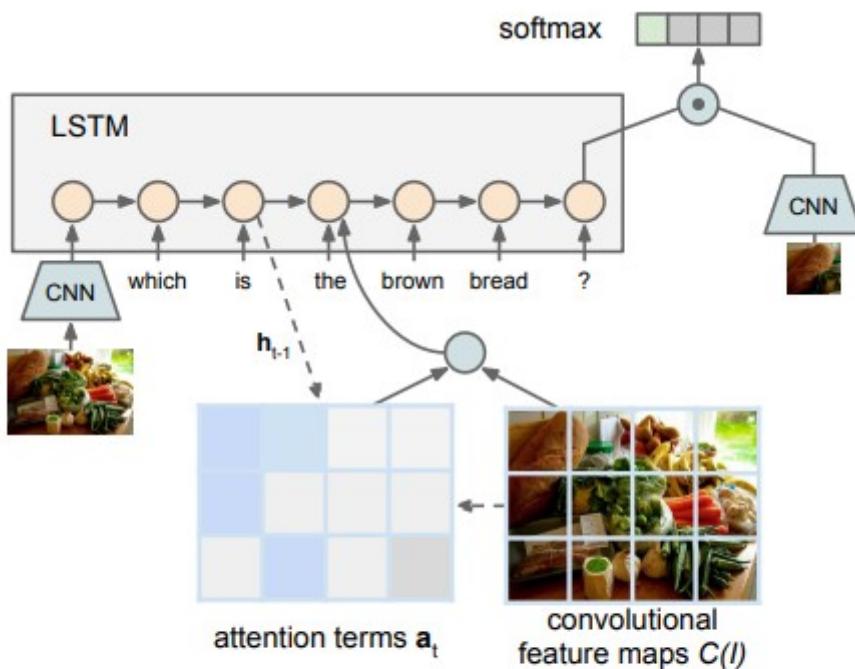
1. 为什么这里bow比lstm好？
2. bin的方式为什么是前两个词？

循环空间注意 Recurrent Spatial Attention (R-SA)

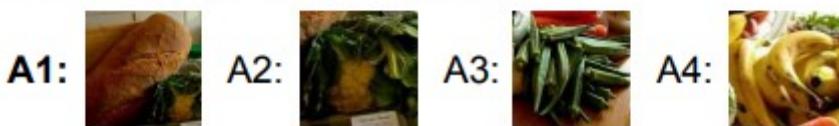
- CNN
- RNN (LSTM)
- Spatial Attention

相关论文：Visual7W: Grounded Question Answering in Images [12:1]

在文中，(Zhu, 2016) 对这个模型的命名为 Recurrent QA Models with Spatial Attention。



Q: Which is the brown bread?



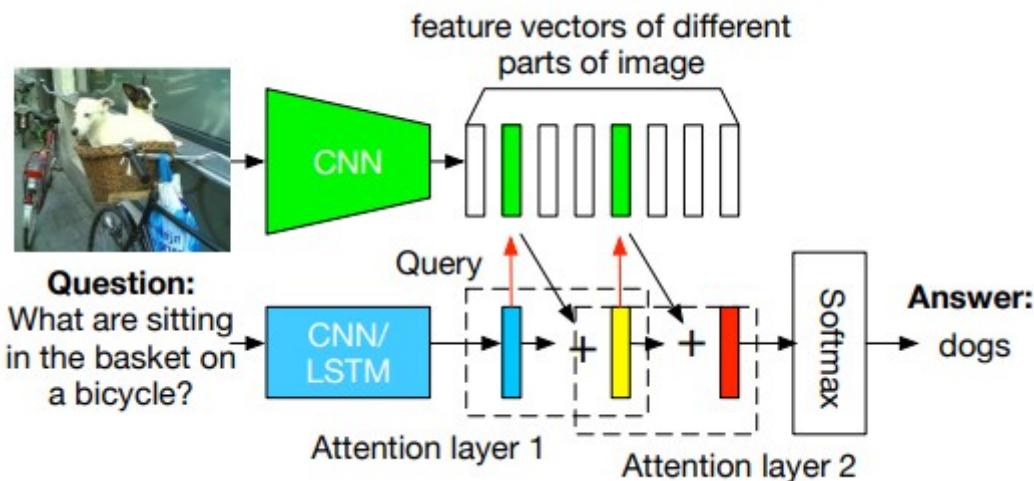
(Zhu, 2016) 在两个方面比上一个模型(WTL)超前一步。首先，它使用 LSTM 对问题进行编码，其次，在扫描问题的每个词之后，它重复地计算图像的注意值。

堆叠注意网络 Stacked Attention Networks (SAN)

- CNN (VGG 19)
- RNN/CNN (LSTM/TextCNN)
- Attention (2 layers)

相关论文：Stacked Attention Networks for Image Question Answering [25]

主要想法：在VQA任务中，按照人为的思路，先定位到自行车，再定位到自行车的兰州，最后看篮子上是什么。这是个推理的过程。所以用分层注意力机制来模拟这个过程。



概括：(Yang, 2016) 采用attention机制来实现这种分层关注的推理过程。在问题特征提取和图像特征提取的思路并没有很特殊，采用LSTM, CNN网络来提取特征。然后用问题特征去attention图像，用attention的结果结合问题向量再次去attention图像，最后产生预测。

图像特征提取：

模型提取 VGG19 最后一个 Pooling 层的 feature map 作为区域特征，其大小为 $14 * 14 * 512$ 。相当于把原始 $448 * 448$ 的图像均匀划分为 $14 * 14$ 个网格 (grid)，每个网格使用一个 512 维的向量表示其特征。(14 * 14是区域的数量，512是每个区域向量的维度，每个feature map对应图像中 $32 * 32$ 大小的区域。)

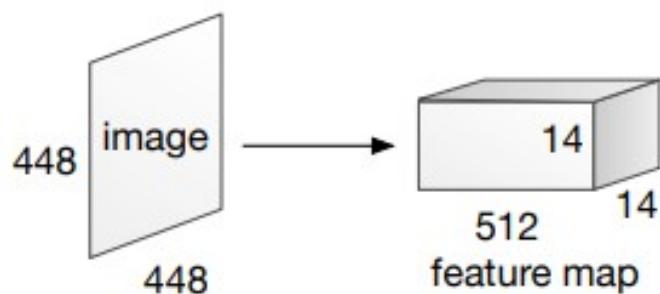


Figure 2: CNN based image model

$$f_I = \text{CNN}_{vgg}(I).$$

问题特征：采用LSTM或者TextCNN。

Stacked Attention:

对于复杂的问题，单一的Attention层并**不足以定位**正确的答案预测区域。本文使用多个Attention层迭代下列过程。



Original Image First Attention Layer Second Attention Layer

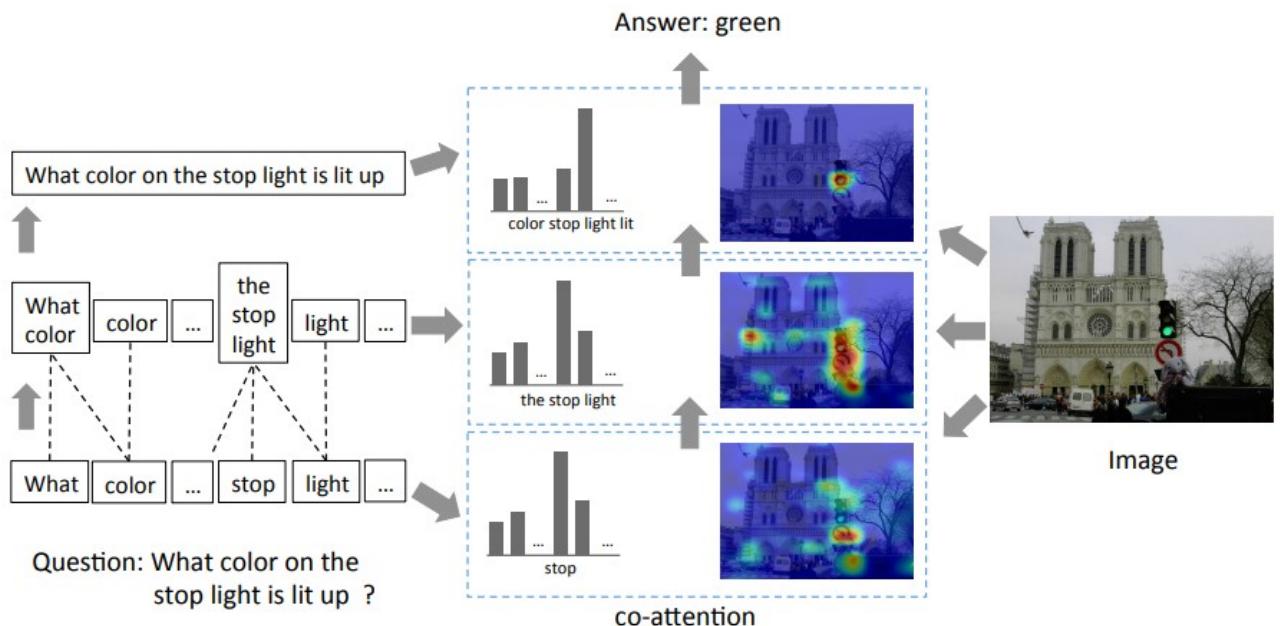
$$\begin{aligned} \text{set } \mathbf{u}^{(0)} &= v_Q, \quad k = 1, 2, \dots, K \\ \text{then } h^{(k)} &= \tanh(W_I^{(k)} \cdot \mathbf{v}_I \oplus (W_Q^{(k)} \cdot \mathbf{u}^{(k-1)} + b_Q^{(k)})) \\ p^{(k)} &= \text{softmax}(W_p^{(k)} \cdot h^{(k)} + b_p^{(k)}) \\ \tilde{\mathbf{v}}_I^{(k)} &= \sum_{i=1}^m p_i^{(k)} \mathbf{v}_i \\ \mathbf{u}^{(k)} &= \tilde{\mathbf{v}}_I^{(k)} + \mathbf{u}^{(k-1)} \end{aligned}$$

本文取 $K=2$

层次协同注意 Hierarchical Co-Attention model

- CNN (VGG/ResNet)
- RNN (LSTM)
- Co-Attention

相关论文：Hierarchical Question-Image Co-Attention for Visual Question Answering [26]



(Lu, 2016) 进一步细化了问题，基于词、短语、句子三个层级分别构建 Attention 权重。

Question Hierarchy:

- word-level feature: 问题映射到一个向量空间，换成词向量
- phrase-level feature: 利用1-D CNN作用于Qw，在每个单词位置计算单词向量和卷积核的内积，卷积核有三个size， unigram, bigram and trigram。
- question-level feature: 将得到的max-pooling结果送入到LSTM中提取特征。全部过程如下图。

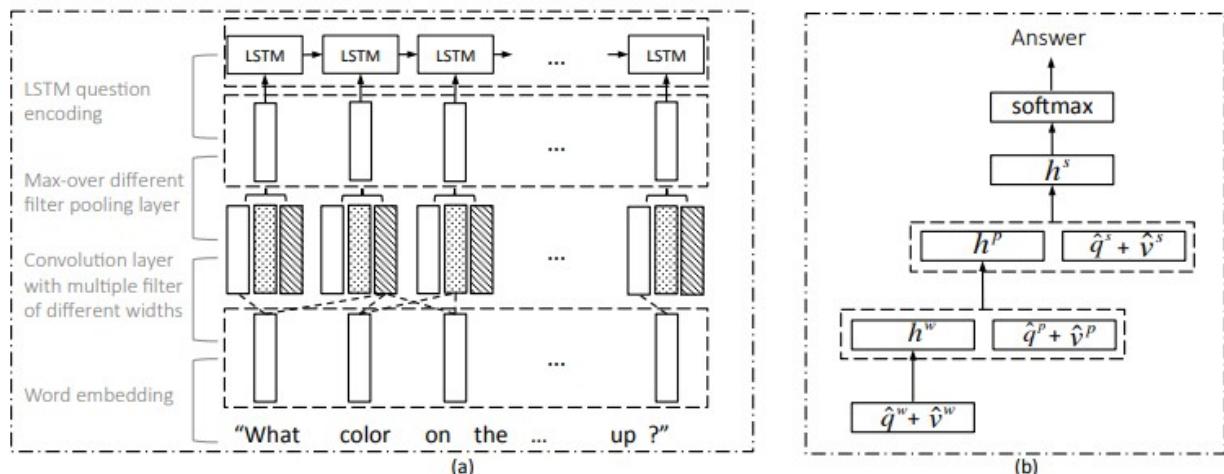


Figure 3: (a) Hierarchical question encoding (Sec. 3.2); (b) Encoding for predicting answers (Sec. 3.4).

两种 Attention 机制：parallel co-attention 和 alternative co-attention：

- parallel co-attention 同时关注问题和图像
- alternative co-attention 同时在关注问题或图像间交替进行
- 最终的答案通过由低到高依次融合三个层级的特征来预测

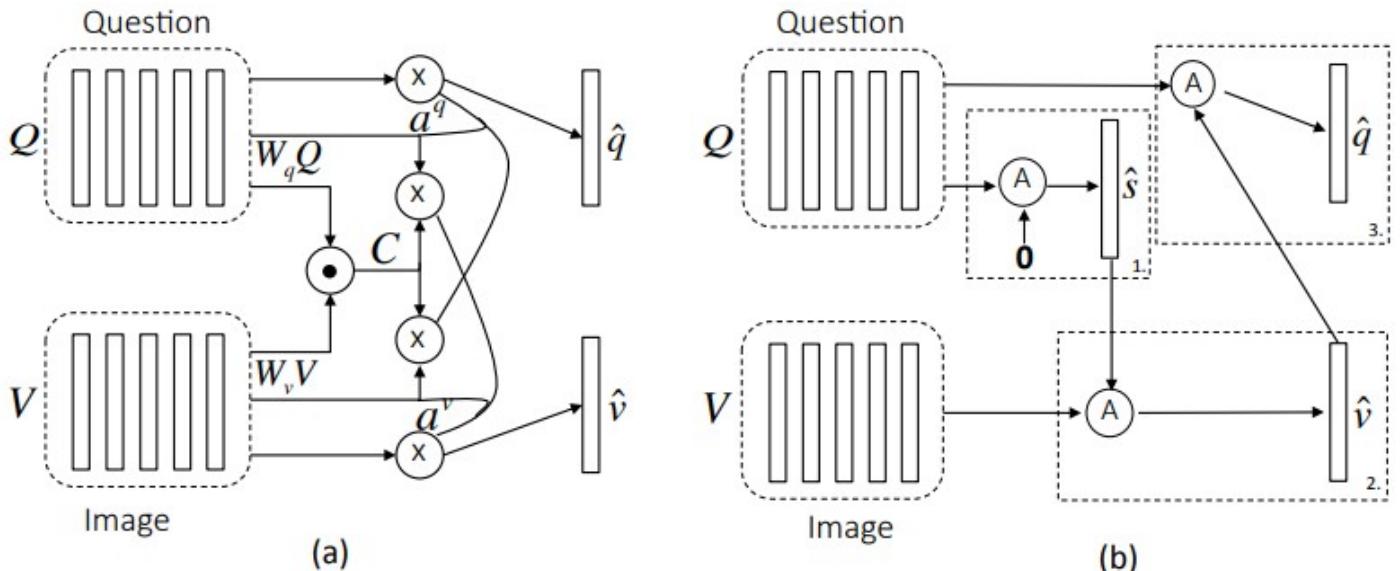


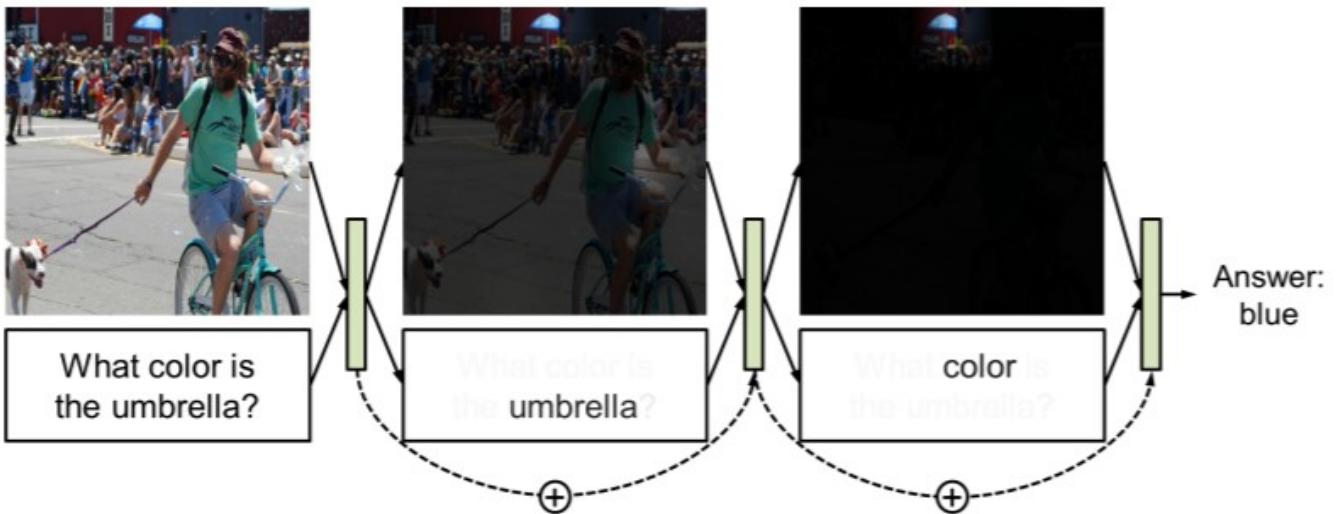
Figure 2: (a) Parallel co-attention mechanism; (b) Alternating co-attention mechanism.

双重注意网络 DAN

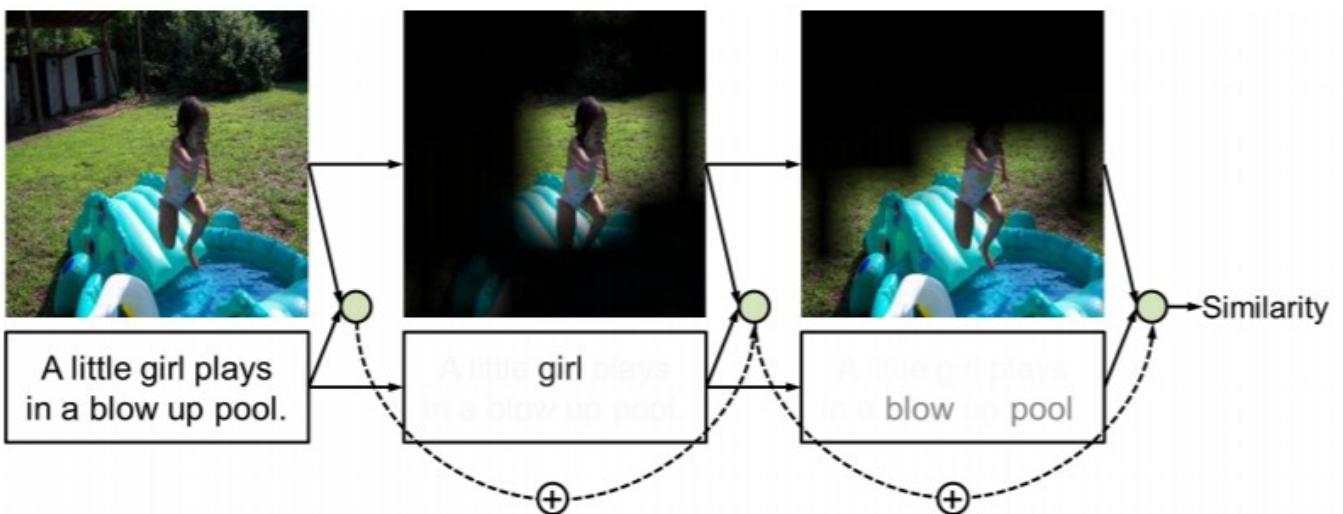
- CNN (VGGNet / ResNet)
- RNN (LSTM)
- Dual Attention

相关论文：Dual Attention Networks for Multimodal Reasoning and Matching [27]

主要思想：(Hyeonseob Nam, 2017) 引入两种类型的DANs (r-DAN用于多模式推理, m-DAN用于多模式匹配) 进行多模态推理, 匹配以及分类。 推理模型允许可视化并在协作推理期间用文本注意机制互相关联。



(a) DAN for multimodal reasoning. (r-DAN)



(b) DAN for multimodal matching. (m-DAN)

Input:

- Image representation 图像特征:

从19层VGGNet 或152层ResNet 中提取的。 我们首先将图像重新缩放到 448×448 并将它们输入到CNN中。为了获得不同区域的特征向量，我们采用VGGNet (pool5) 的最后一个池化层或ResNet最后一个池化层 (res5c) 的下面一层。

- Text representation 文本特征:

使用双向LSTM来生成文本特征：提取出T个文本特征

Attention Mechanisms:

- Visual Attention: 分别将初始化的图像特征向量(在r-DAN中为前一层的memory vector即前一层图像特征与文本特征的点乘)和图像的特征用两层前馈神经网络(FNN) 相连，然后再用tanh激活并做点乘，然后用softmax做归一化得到权重向量 (N维向量)，利用权重向量将N个2048维的向量做加权平均，然后再乘以一个权重矩阵，最后再用tanh进行激活，得到图像attention向量。

- Textual Attention: 将初始化的文本特征向量query(在r-DAN中为前一层的memory vector即前一层图像特征与文本特征的点乘)和文本的特征 key 用两层前馈神经网络(FNN) 相连，然后再用tanh激活并做点乘，然后用 softmax 做归一化得到权重向量 (N维向量)，利用权重向量将N个512维的向量做加权平均，得到文本attention向量。

DAN:

解决了两种不同的问题，都用到了前面的Attention机制，但是不同的问题，提出了r-DAN(用于VQA)和m-DAN (用于Image-Text Matching) 两种模型.

- r-DAN for Visual Question Answering

VQA本质上为**分类问题**，将图像attention特征和文本attention特征融合得到memory vector，做分类任务。

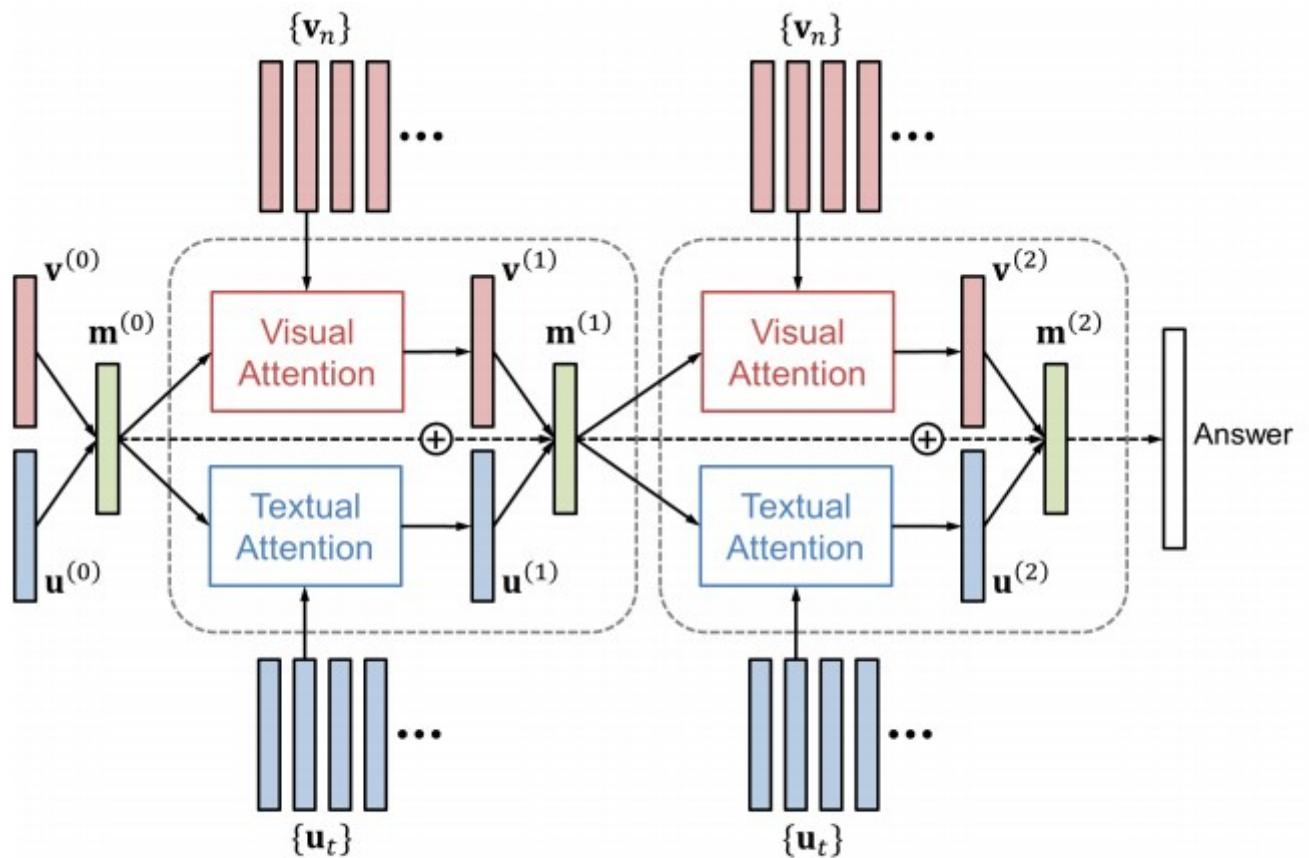


Figure 3: r-DAN in case of $K = 2$.

- m-DAN for Image-Text Matching

图文匹配问题与VQA最大的不同就是，他要解决的是一个**Rank问题**，所以需要比对两种特征之间的距离，因此就**不能共享一个相同的Memory Vector**。

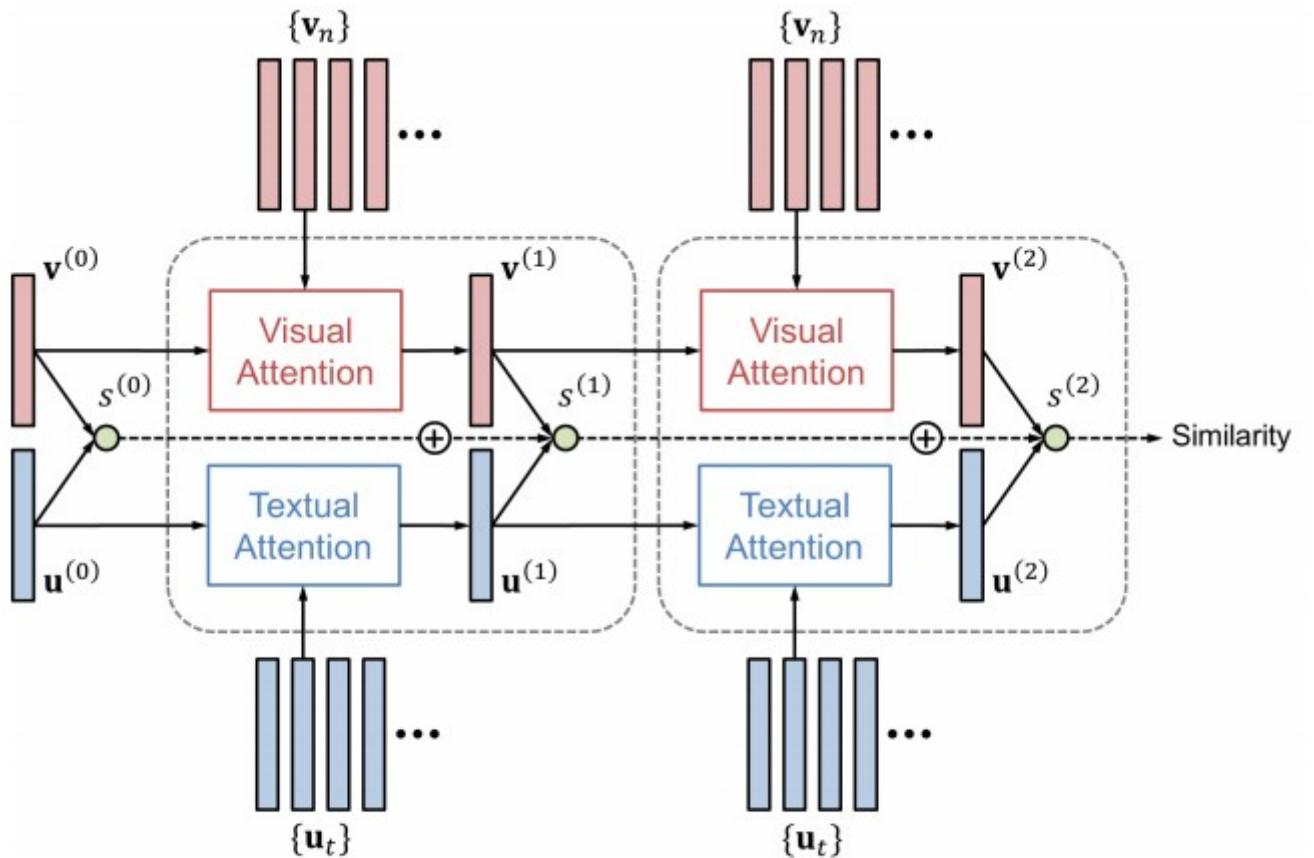


Figure 4: m-DAN in case of $K = 2$.

Loss Function: Triplet Loss (文章中没有提到hard的思想，负样本应该是在minibatch里面随机选的)

Tips and Tricks for Visual Question Answering

- Faster-RCNN
- Glove Vectors

相关论文：Tips and Tricks for Visual Question Answering: Learnings from the 2017 Challenge [28]

这是一篇很工程实践性质的论文。

本篇文章作者 (D Teney, 2017) 获得了2017 VQA Challenge的第一名，花费了3000小时的GPU运算。为了获得第一名，文中使用了很多技巧来提升性能，但核心出发点都要依赖joint embedding和multi-label classifier方法来解决VQA问题的建模，换句话说就是利用视觉特征和语义特征进行有效融合，然后依赖特征在候选答案上做multi-label预测（区别于softmax多类预测，形象比喻就是softmax最后得到的是N类的预测向量，而multi-label可以认为是得到预测矩阵，每一行表示对应问题答案的预测向量，当然这只是比喻，并不严谨）。简单说，multi-label的通常实现方式有两种，一种是 SigmoidCrossEntropyLoss，另一种是使用多个SoftmaxWithLoss。

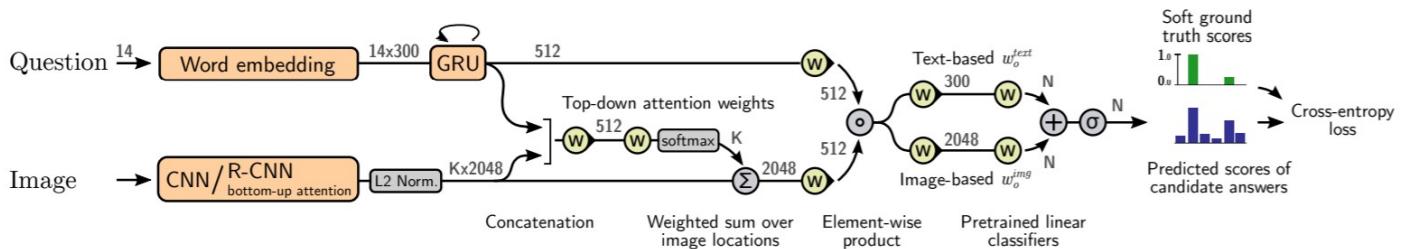


Figure 2. Overview of the proposed model. A deep neural network implements a joint embedding of the input question and image, followed by a multi-label classifier over a fixed set of candidate answers. Gray numbers indicate the dimensions of the vector representations between layers. Yellow elements use learned parameters. The elements \textcircled{W} represent linear layers, and $\textcircled{\textcircled{W}}$ non-linear layers (gated tanh).

所用到的关键技巧主要有：

1. 使用sigmoid outputs来从每个问题中的允许多个答案，替代single-label softmax
2. 使用soft scores as ground truth targets用回归代替分类
3. 使用image features from bottom-up attention来针对感兴趣区域提特征，替代之前grid-like的方法
4. 使用gated tanh activations作为激活函数
5. 使用pretrained representations of candidate answers初始化输出layer的权重
6. 使用large mini-batches and smart shuffling of training data来训练

Question embedding: 采用GRU进行编码问题

词向量采用GloVe词向量（300维）；词向量中没有的初始化为0；文本长度用14截断；GRU内部状态为512。

Image features: 图像特征，有两种方式

- 直接用cnn：使用预训练的ImageNet，比如说，200-layer ResNet，得到772048
- bottom-up attention：使用Faster R-CNN framework提取图像中的topk目标。k可以调节，最大取100。

Image attention: 图像的attention，当然了还可以考虑多次attention、stack等

Multimodal fusion: 多模态特征融合joint embedding，采用对应位置相乘的方式，即Hadamard product。

Output classifier: 把候选答案结合作为输出词典，通过正确答案在训练集上出现8次的，放入输出词典中（N=3129）。由于标注的模糊性，训练集有7%的问题没有正确答案。实验也发现，对于这是模糊的问题，multi-label几乎没有预测输出。

Petraining the classifier: 预训练分类器（分类网络初始化）

由于分类网络最后一层是个全连接层，所以最后每个答案的分数就是图片特征和问题特征与网络权重的点积。

作者使用了来自两个来源的候选答案的先验信息来训练：

- 一种是语言信息，使用答案文本的GloVe词嵌入形式的语言信息，当一个答案不能与问题完全匹配时，在拼写检查后则选择关系程度最接近的匹配，删除连字符号，或者保留多词表达式中的单个词，矩阵W0(text)(语言信息)的每一行通过答案的 glove feature 进行初始化；

- 一种是视觉信息，是从代表候选答案的图像中收集的视觉信息。利用Google Images来自动检索挑选10张与每个候选答案最接近的图像。这些图片经过一个在ImageNet上预训练过的ResNet-101 CNN，最终的平均特征被提取并在这10张图片上取平均值，用它作为W0(img)(视觉信息)每一行的初始化（2048维）；

两种先验信息W0(text)与W0(img)互补结合，它们可以用于任何候选答案，包括多义词和生僻词

相关论文 2: Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering [29]

Github: <https://github.com/markdtw/vqa-winner-cvprw-2017>

Pythia v0.1

极致的工程

相关论文：Pythia v0.1: the Winning Entry to the VQA Challenge 2018 [30]

Pythia 以 VQA 2017 Challenge 的冠军模型 Up-Down 为基本方法，辅助以了诸多工程细节上的调整，这使得 Pythia 较往年增加了约 2% 的性能提升 ($70.34\% \rightarrow 72.25\%$)。

Model	test-dev	test-std
up-down [1]	65.32	65.67
up-down Model Adaptation (§2.1)	66.91	
+ Learning Schedule (§2.2)	68.05	
+ Detectron & Fine-tuning (§2.3)	68.49	
+ Data Augmentation* (§2.4)	69.24	
+ Grid Feature* (§2.5)	69.81	
+ 100 bboxes* (§2.5)	70.01	70.24
Ensemble, $30 \times$ same model (§2.6)	70.96	
Ensemble, $30 \times$ diverse model (§2.6)	72.18	72.27

模型结构：65.32% → 66.91%

- 还记得 Up-Down 里面那个长相奇怪的门控激活函数吗？Pythia 使用了 RELU+Weight Normalization 来取代它，这样可以降低计算量，但是效果上有无提升文中没有给出实验。
- 在进行 top-down 的 attention 权重计算时，将特征整合的方式由原本 concat 转换为 element-wise multiplication，这也是可以降低计算量的表现。

- 在第二个 LSTM 做文本和图像的联合预测时，hidden size 为 5000 最佳。

超参数：66.91% → 68.05%

这里主要是学习率的调整。作者发现在 Up-Down 模型中适当减小 batch 可以带来一些提升，这意味着在同样的 batch 下提升学习率可能带来性能的提升。为了防止学习率过大不收敛，他们采用了广泛使用的 warm-up 策略，并使用了适当的 lr step。

Faster R-CNN 增强：68.05% → 68.49%

将 Faster R-CNN 的 backbone 由 ResNet-101 换为 ResNext-101-FPN，并且不再使用 ROI Pooling 后的 $7 \times 7 \times 2048 + \text{mean pooling}$ 表征 object-level 特征，而采用 fc7 出来的 2048 维向量以减少计算量。

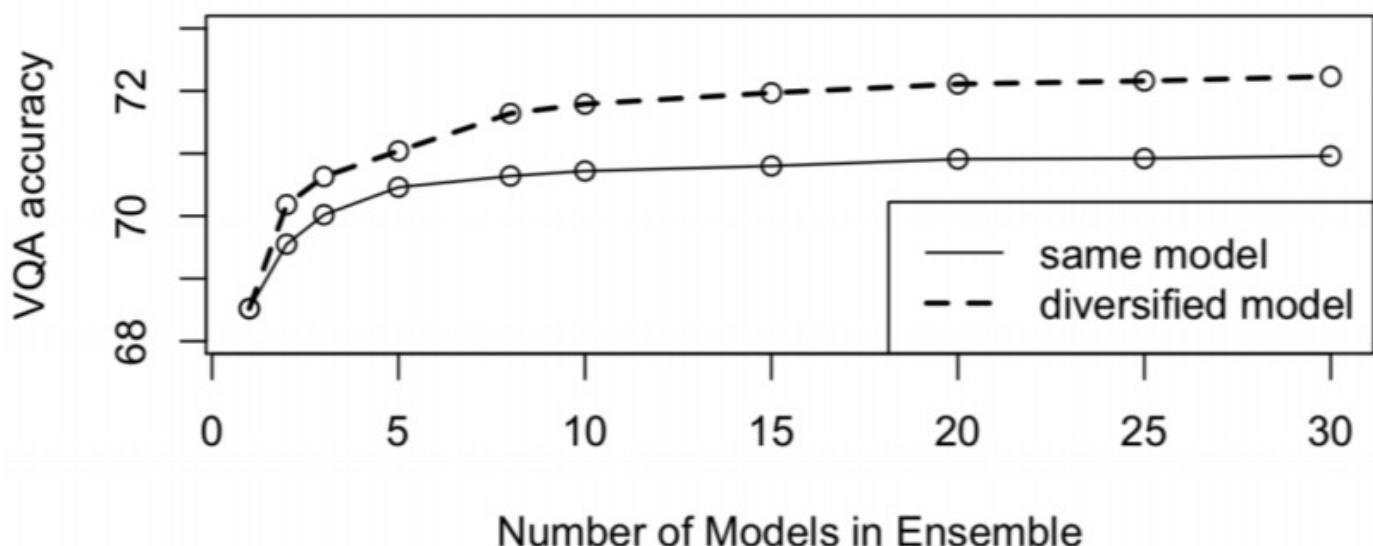
数据增强：68.49% → 69.24%

采用了图像水平翻转的增强方法，这样的方式在纯视觉任务中广泛出现。在这里还需要做变换的是，将问题和答案中的“左”和“右”对调。

Bottom-up 增强：69.24% → 70.01%

光是使用 Faster R-CNN 在 head network 上的 fc7 特征不足以表示图像整体的特征。于是作者们融合了 ResNet-152 提取的整图特征，并且增加了在每一张图提取 object-level feature 的个数。它们分别带来了可见的提升。

模型集成：70.96% → 72.18%



Github 1: <https://github.com/gabegrand/pythia-1>
Github 2: <https://github.com/meetshah1995/pythia-1>

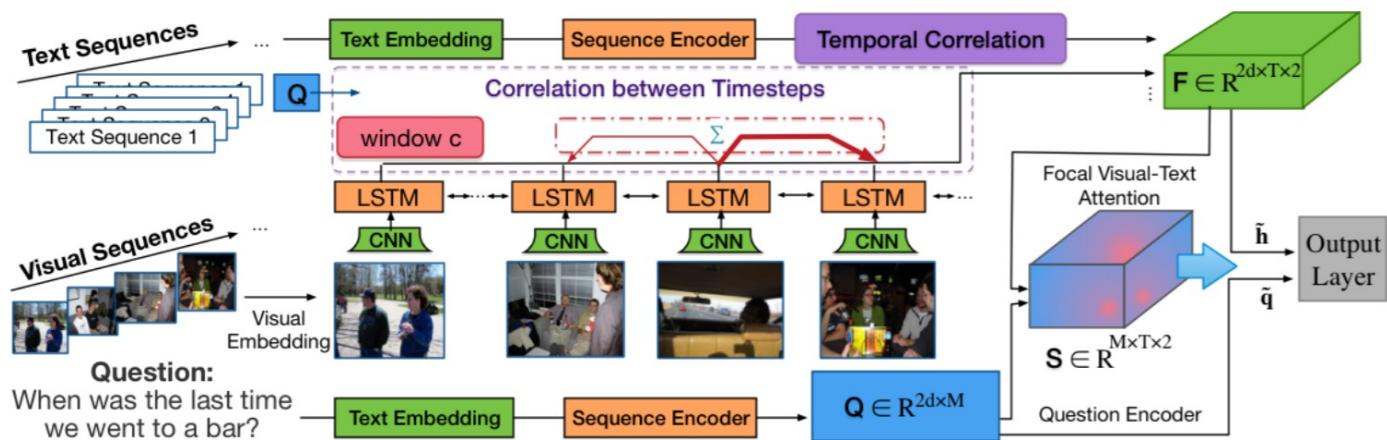
Focal Visual-Text Attention (FVTA)

这项工作 (J Liang, 2018) 在两个方面不同于现有的基于视频的问答:

- (1) 基于视频的问答是基于单个视频回答问题, 而这个工作可以处理一般的可视文本序列, 其中一个用户可能有多个视频或相册。
- (2) 大多数现有的基于视频的质量保证方法将一个带有文本的视频序列映射到一个上下文特征向量中, 而这篇文章通过在每个时间步建模查询和序列数据之间的相关性来探索一个更细粒度的模型。

这项工作可以被视为一个新的关注模型, 为多个可变长度的顺序输入, 不仅考虑到视觉文本信息, 还考虑到时间的依赖性。

相关论文: Focal Visual-Text Attention for Visual Question Answering [31]



模型结构:

Visual-Text Embedding 每个图像或视频帧都用预先训练的卷积神经网络编码。单词级和字符级嵌入都被用来表示文本和问题中的单词。

Sequence Encoder 使用独立的LSTM网络分别对视觉和文本序列进行编码, 以捕获每个序列中的时间相关性。LSTM单元的输入是由前一层产生的图像/文本嵌入。

Focal Visual-Text Attention FVTA 是实现所提出的注意机制的一个新层。它表示一个网络层, 该层对问题和多维上下文之间的相关性进行建模, 并将汇总后的输入输出到最终的输出层。

Output Layer 在使用FVTA注意力总结输入之后, 使用前馈层来获得候选答案。

Github: https://github.com/JunweiLiang/FVTA_MemexQA

3.4 其它有趣的模型

不同于前面的模型, 下面的模型使用了更多的思想, 而不仅仅是在计算图像或问题的注意值方面作改变。

MCBP for VQA

相关论文：Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding [32]

Bilinear pooling在2015年于 "Bilinear CNN Models for Fine-grained Visual Recognition" [33] 被提出来用于fine-grained分类后，又引发了一波关注。bilinear pooling主要用于特征融合，对于从同一个样本提取出来的特征x和特征y，通过bilinear pooling得到两个特征融合后的向量，进而用来分类。

(A Fukui, 2016) 在CBP的基础上提出了MCBP。

注意到CBP是针对HBP进行改进的，对CBP的TS算法稍加改动，使其适用于融合不同模态的特征，即可得到MCBP，如下图所示。

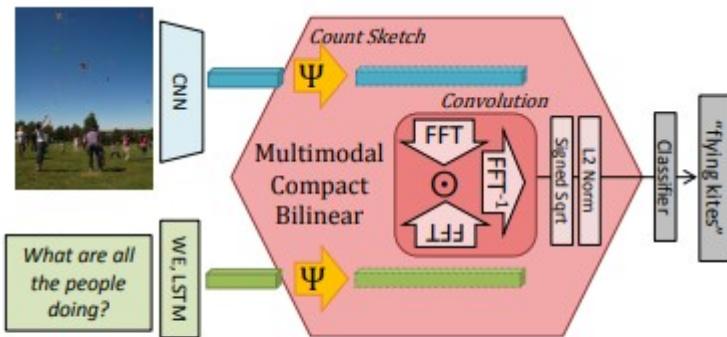
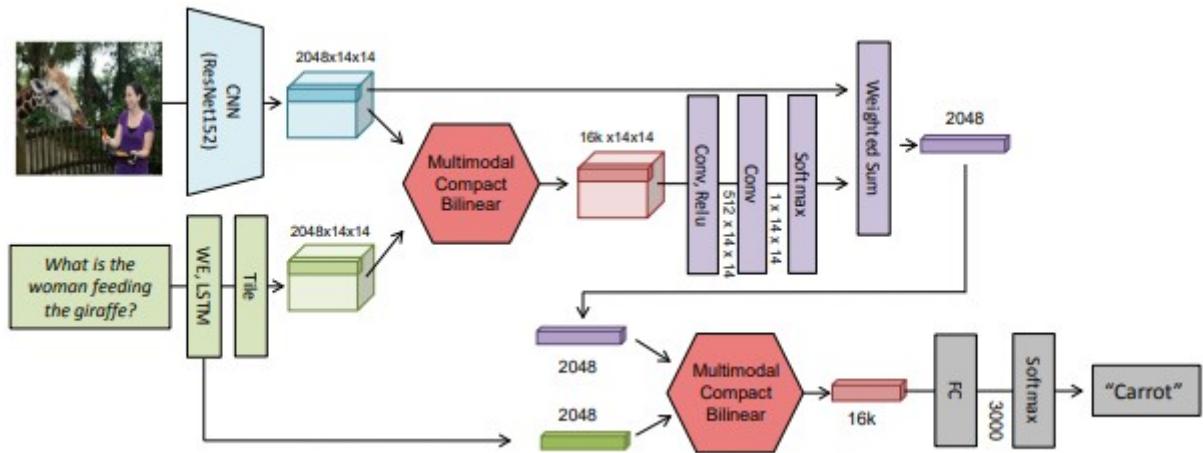


Figure 1: Multimodal Compact Bilinear Pooling for visual question answering.

文本计算 Attention 的做法类似，区别在于使用 MCB 操作代替双线性 Attention。在得到MCBP模块后，作者提出用于VQA的网络结构如下：



这里用到了两次MCB模块，第一个MCB融合图像特征和文本特征计算图像每个空间位置的attention weight。第二个MCB融合图像特征和文本特征得到答案。

本文模型是 2016 VQA 比赛的获胜模型。

Github 1: https://github.com/gdlg/pytorch_compact_bilinear_pooling

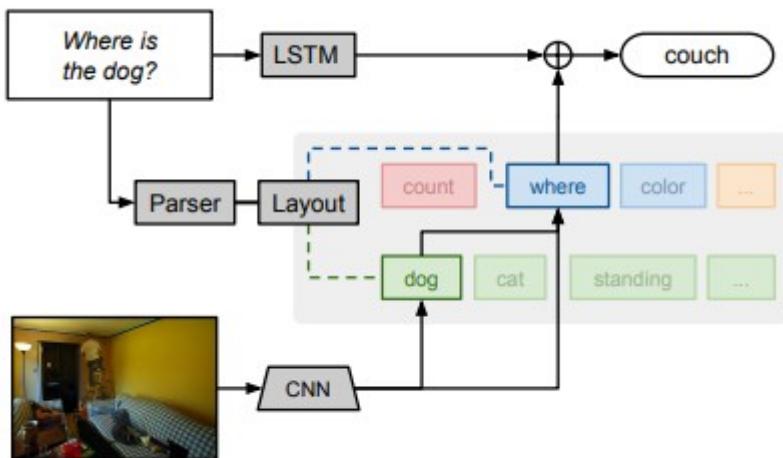
Github 2: <https://github.com/akirafukui/vqa-mcb>

Github 3: <https://github.com/jnhwkim/cbp>

神经模块网络 Neural Module Network (NMN)

相关论文：Neural Module Networks [34]

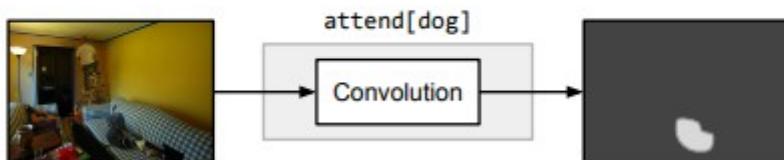
(J Andreas, 2015) 提出的NMN的一大特点就是其结构是它并不是像传统的神经网络模型一样是一个整体，它是由多个模块化网络组合而成。根据VQA数据集中每个questions定制一个网络模型。也就是说NMN模型的网络是根据question的语言结构**动态生成**的。



有五种模块：Attention, Re-attention, Combination, Classification 和 Measurement。

- Attention

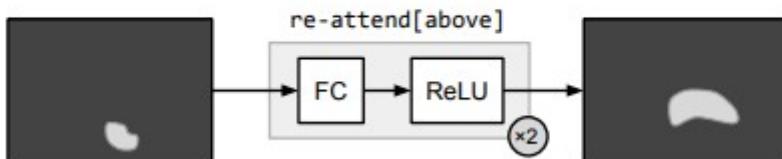
$$\text{attend} : \text{Image} \rightarrow \text{Attention}$$



attend模块将输入图像的每一个位置与权重（根据C的不同而不同）提供一个热力图或一个非标准的注意力图。比如，attend[dog]模块输出的矩阵，包含狗的区域值较大，而其他区域值较小。

- Re-attention

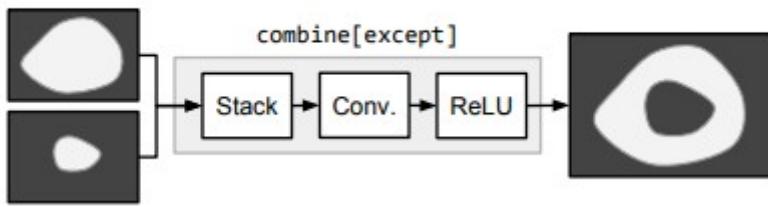
$$\text{re-attend} : \text{Attention} \rightarrow \text{Attention}$$



Re-attention模块本质上由多元感知器及Relu实现，执行一个全连接使得将注意力映射到其他地方。re-attend[above]就是讲attention和最佳的软激活区域向上移。

- Combination

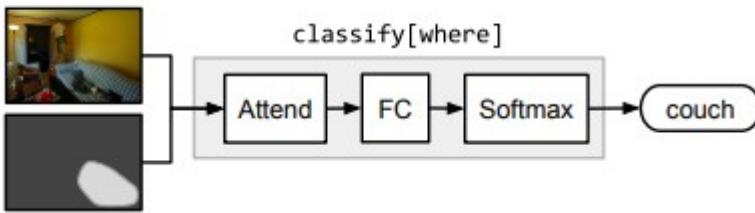
$\text{combine} : \text{Attention} \times \text{Attention} \rightarrow \text{Attention}$



`combine`模块将两个attention结合成一个attention。比如，`combine`只激活两个输入中都激活的区域，而`except`则是激活第一个输入，将第二个输入失活。

- Classification

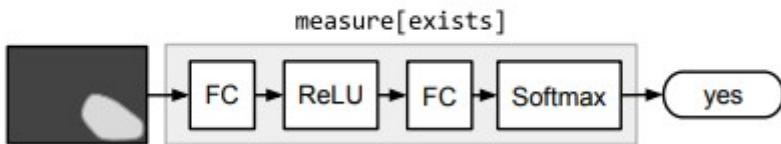
$\text{classify} : \text{Image} \times \text{Attention} \rightarrow \text{Label}$



`Classification`模块`classify`将attention和image映射到labels的概率分布。它首先计算由注意力加权的平均图像特征，然后通过一个完全连通层传递这个平均特征向量。

- Measurement

$\text{measure} : \text{Attention} \rightarrow \text{Label}$



`Measure`模块`Measure[c]`以一个attention作为输入，映射到label的概率分布。由于传递的attention是非标准的，所以`measure`模块适合用于评价检测目标是否存在。

网络结构的生成：

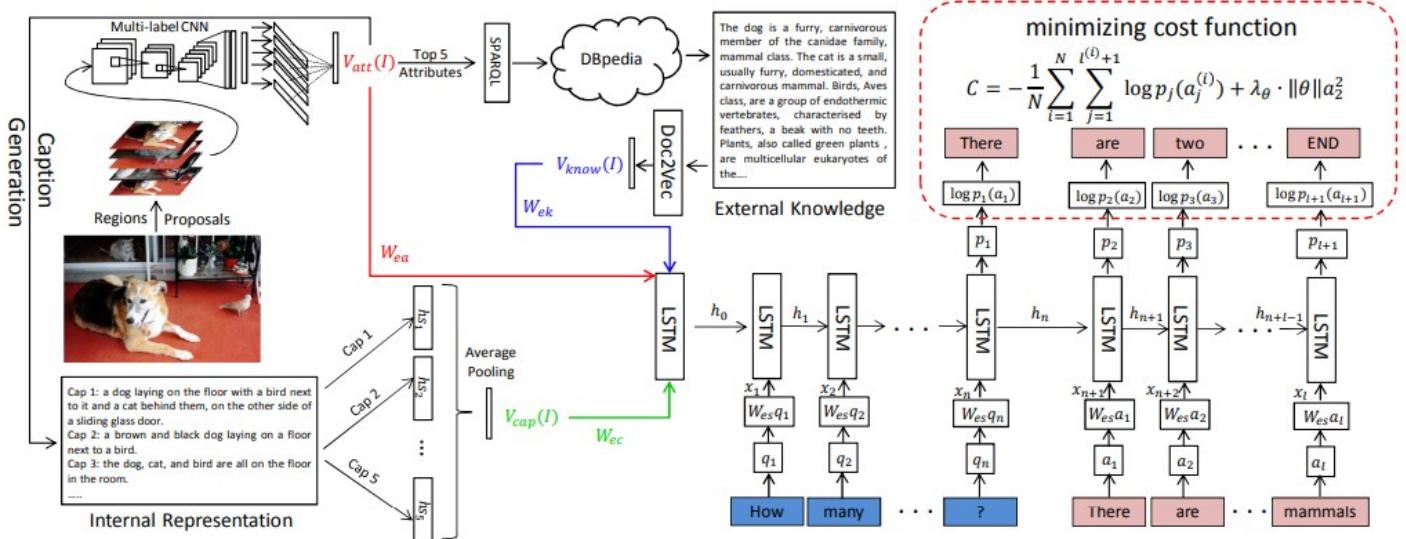
已经建立了模块集合，就需要将它们根据不同问题组装成不同的网络布局。从自然语言问题到神经网络实例化有两个步骤。

- 将自然语言问题映射成布局 (layouts)
- 使用布局(layouts)组建最终的预测网络

AMA based on KB

相关论文：Ask Me Anything: Free-form Visual Question Answering Based on Knowledge from External Sources [35]

(Wu, 2016) 提出了 Ask Me Anything (AMA)模型，该模型试图借助外部知识库中的信息来帮助指导视觉问答。将自动生成的图像描述与一个外部的Knowledge bases相融合，对问题进行预测。图像描述生成主要来自于image captions集，并且从Knowledge bases提取基于文本的外部知识。



总体上看大致分为这样4个步骤：

- 先从图像中提取前五的属性。
- 提取的属性分为三部分：一方面用来直接生成关于图像的描述，另一方面用来从知识库中提取相关外部知识，当然，自身也会被重新用到。
- 将第二步中的图像的三个结果作为一个视觉信息的整体输入到LSTM的编码结构中，问题的每个单词也作为输入输入到LSTM的编码结构中。然后在LSTM的解码结构中，生成每个答案单词的分布概率。
- 最终得到一个多个单词标签的答案。

缺点在于仅仅从数据集中提取离散的文本描述，忽略了结构化的表达，也就是说，没有办法进行关系推理，没有说明为什么是这个外部知识，从数据库中找到仅仅是相关的描述。

NS-VQA

- 所有module都基于Attention

相关论文：Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding [36]

主要思想：(K Yi, 2018) 提出的神经符号视觉问答 (NS-VQA) 系统首先会根据图像恢复一个结构化的场景表征，并会根据问题恢复一个程序轨迹。然后它会在这个场景表征上执行该程序以得到答案。

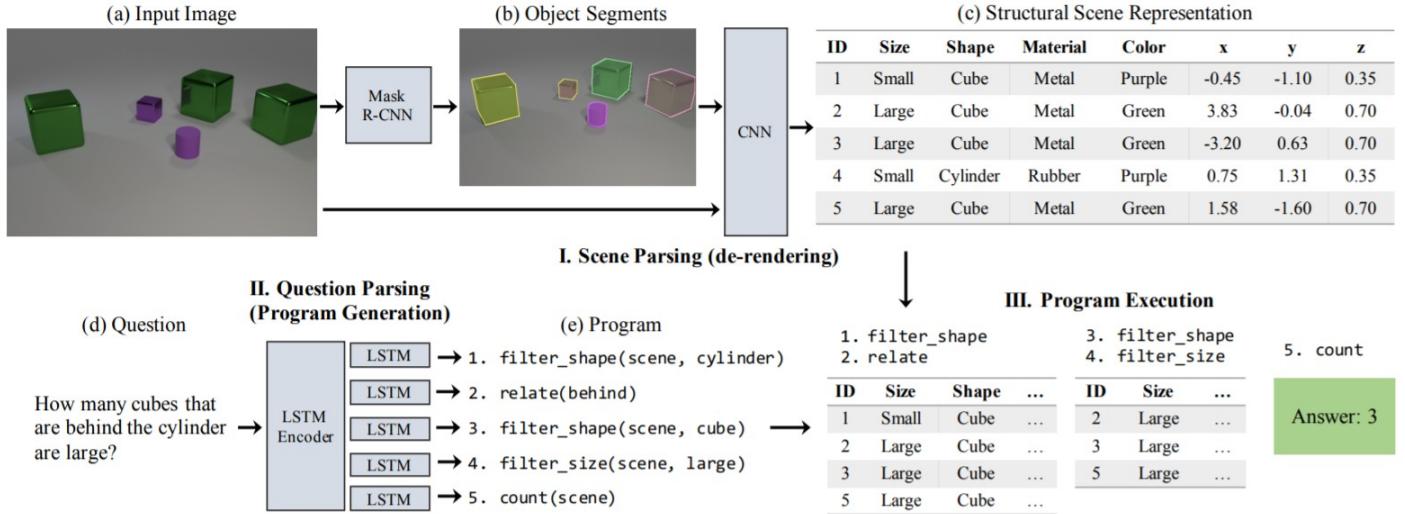


Figure 2: Our model has three components: first, a scene parser (de-renderer) that segments an input image (a-b) and recovers a structural scene representation (c); second, a question parser (program generator) that converts a question in natural language (d) into a program (e); third, a program executor that runs the program on the structural scene representation to obtain the answer.

NS-VQA 模型有三个组件：场景解析器（去渲染器/de-renderer）、问题解析器（程序生成器）和程序执行器。给定一个图像-问题对，场景解析器会去除图像的渲染效果，得到结构化的场景表征（I），问题解析器会基于问题生成层次化的程序（II），程序执行器会在结构化的表征上运行程序从而得到答案（III）。

优点：

- 符号表征的使用能提供对长的复杂程序轨迹的稳健性。它还能减少对训练数据的需求。
- 推理模块和视觉场景表征都是轻量级的，仅需要最少量的计算和内存成本。
- 符号场景表征和程序轨迹的使用能迫使模型准确地基于问题恢复底层的程序。结合完全透明且可解读的符号表征的本质，可以对推理过程进行一步步地分析和诊断。

Github: <https://github.com/kexinyi/ns-vqa>

差分网络 Differential Networks

(B Patro, 2018) 提出通过一或多个支持和反对范例来取得一个微分注意力区域 (differential attention region)，与基于图像的注意力方法比起来，本文计算出的微分注意力更接近人类注意力，因此可以提高回答问题的准确率。

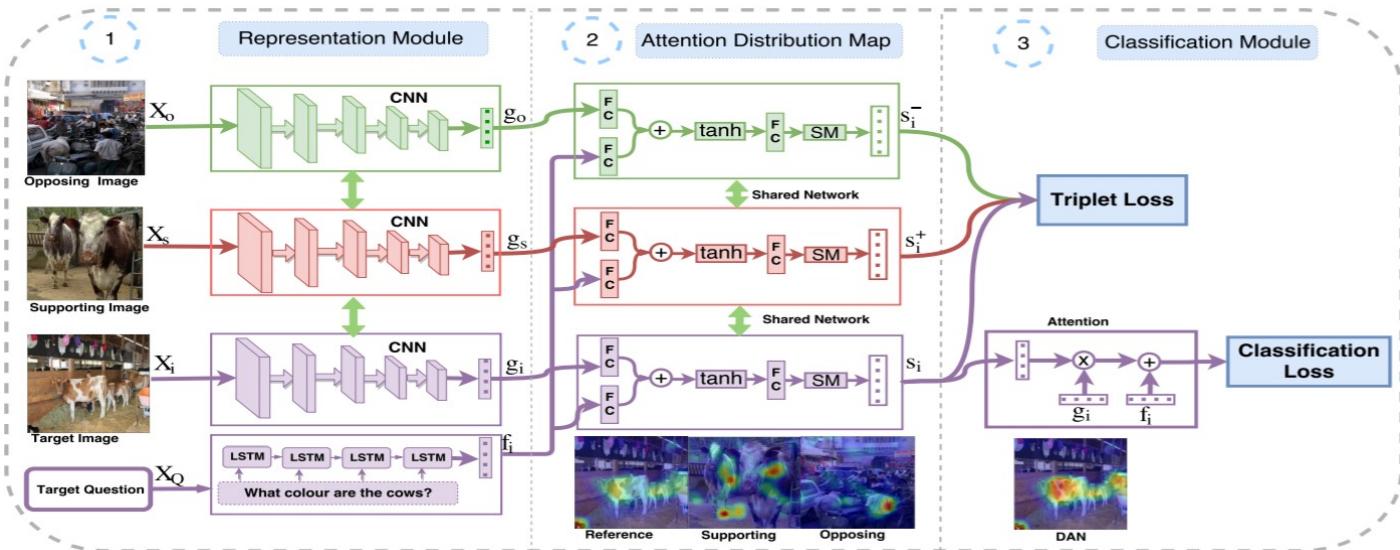
相关论文：Differential Attention for Visual Question Answering [37]

原理流程：

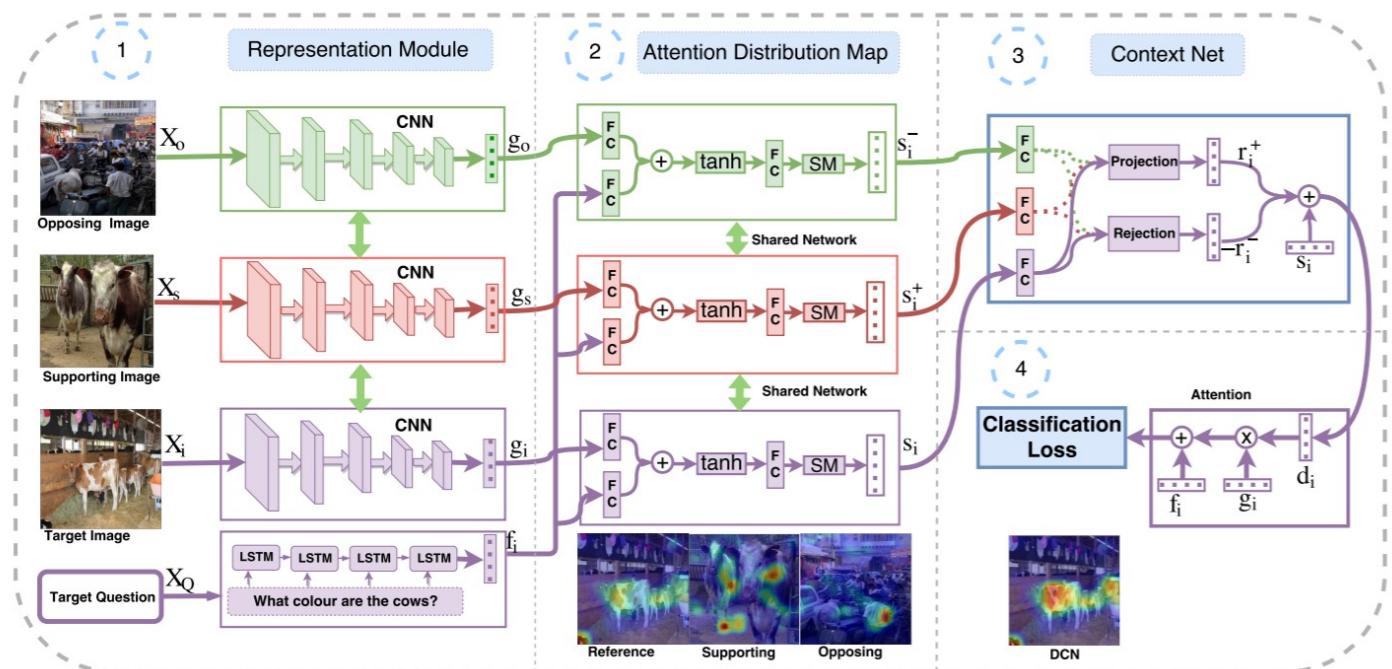
1. 根据输入图像和问题取得引用注意力嵌入 (reference attention embedding)；
2. 根据该引用注意力嵌入，在数据库中找出样本，取近样本作为支持范例、远样本作为反对范例；
3. 支持范例和反对范例用于计算微分注意力向量；

4. 通过微分注意力网络 (differential attention network, DAN) 或微分上下文网络 (differential context network) 分别可以改进注意力或取得微分上下文特征，这两种方法可以提升注意力与人工注意力的相关性；

首先为微分注意力网络 (differential attention network, DAN) ，重点为通过正反例注意力更新目标标注注意力，使之与人类的注意力更相似。



然后就是微分上下文注意力(DCN)，其主要应用映射的概念，缩小正例与目标注意力之间的距离，删除反例上下文与目标注意力之间的特征，从而达到更新注意力的目的。



创新点是引入了支持示例和相对示例进而找到与答案相关的区域，进行回答问题。

3.5 暂时的小结

下面两张图是上述部分模型在部分数据集上的表现。

	DAQUAR (Reduced)			DAQUAR (All)			COCO-QA		
	Accuracy (%)	WUPS at 0.9 (%)	WUPS at 0 (%)	Accuracy (%)	WUPS at 0.9 (%)	WUPS at 0 (%)	Accuracy (%)	WUPS at 0.9 (%)	WUPS at 0 (%)
SWQA	9.69	14.73	48.57	7.86	11.86	38.79	-	-	-
MWQA	12.73	18.10	51.47	-	-	-	-	-	-
Vis+LSTM	34.41	46.05	82.23	-	-	-	53.31	63.91	88.25
AYN	34.68	40.76	79.54	21.67	27.99	65.11	-	-	-
2Vis+BLSTM	35.78	46.83	82.15	-	-	-	55.09	65.34	88.64
Full-CNN	42.76	47.58	82.60	23.40	29.59	62.95	54.95	65.36	88.58
DPPnet	44.48	49.56	83.95	28.98	34.80	67.81	61.19	70.84	90.61
ATP	45.17	49.74	85.13	28.96	34.74	67.33	63.18	73.14	91.32
SAN	45.50	50.20	83.60	29.30	35.10	68.60	61.60	71.60	90.90
CoAtt	-	-	-	-	-	-	65.40	75.10	92.00
AMA	-	-	-	-	-	-	69.73	77.14	92.50

Table 2: Results of various models on DAQUAR (reduced), DAQUAR (full), COCO-QA

	Test-Development					Test-Standard				
	Open Ended			M.C.	All	Open Ended			M.C.	All
	Y/N	Number	Other			All	All	All		
iBOWIMG	76.5	35.0	42.6	55.7	-	76.8	35.0	42.6	55.9	-
DPPnet	80.7	37.2	41.7	57.2	-	80.3	36.9	42.2	57.4	-
WTL	-	-	-	-	62.4	-	-	-	-	62.4
AYN	78.4	36.4	46.3	58.4	-	78.2	36.3	46.3	58.4	-
SAN	79.3	36.6	46.1	58.7	-	-	-	-	58.9	-
ATP	80.5	37.5	46.7	59.6	-	80.3	37.8	47.6	60.1	-
NMN	81.2	38.0	44.0	58.6	-	81.2	37.7	44.0	58.7	-
CoAtt	79.7	38.7	51.7	61.8	65.8	-	-	-	62.1	66.1
AMA	81.01	38.42	45.23	59.17	-	81.07	37.12	45.83	59.44	-

Table 3: Results of various models on VQA dataset

有趣的是，我们看到 ATP 模型的表现优于非注意模型，这证明简单地引入卷积和/或循环神经网络是不够的：原则上识别相关的图像部分是重要的。ATP 甚至可以与一些注意模型（如 WTL 和 SAN）相媲美甚至表现更好。

CoAtt 的表现有显著的提升，该模型首先注意问题然后注意图像。这对于长问题可能是有帮助的，由于这些问题更难用 LSTM/GRU 编码表示为单个向量，因此首先对每个词进行编码，然后使用图像来注意重要的词，这样有助于提高模型的准确率。NMN 模型使用了为每个（图像/问题）对自动组合子模型的新颖想法，它的表现效果类似于在 VQA 数据集上的 CoAtt 模型，但是在需要更高级推理的合成数据集上优于所有模型，表明该模型在实际中可能是一种有价值的方法。然而，需要更多的测试来判断该模型的性能。

在 COCO-QA 数据集上表现最好的模型是 AMA 模型，它包含外部知识库（DBpedia）的信息。这样做的一个可能的原因是知识库帮助解决涉及常识的问题，而这些知识可能不在数据集中。该模型在 VQA 数据集上的表现不是很好，这可能是因为这个数据集没有太多的问题需要常识。自然地这种模型会为未来的工作带来两大方向。第一个方向是认识到外部知识的必要性：某种 CoAtt 和 AMA 的混合模型加上是否访问知识库的决策器可能会兼有两种模型的优点。该决策器可能是面向应用的，以实现端到端的训练。第二个方向是探索使用其它知识库，如 Freebase、NELL 或 OpenIE 的信息提取。

Model	Dataset(s)	Method	Accuracy	Venue
Vanilla VQA [1]	VQA [1]	CNN + LSTM	54.06 (VQA)	ICCV 2015
Stacked Attention Networks [12]	VQA [1], DAQUR [8], COCO-QA [21]	Multiple Attention Layers	58.9 (VQA), 46.2 (DAQUR), 61.6 (COCO-QA)	CVPR 2016
Teney et al. [14]	VQA [1]	Faster-RCNN [22] + Glove Vectors [23]	63.15 (VQA-v2)	CVPR 2018
Neural-Symbolic VQA [24]	CLEVR [17]	Symbolic Structure as Prior Knowledge	99.8 (CLEVR)	NIPS 2018
FVTA [25]	MemexQA [26], MovieQA [27]	Attention over Sequential Data	66.9 (MemexQA), 37.3 (MovieQA)	CVPR 2018
Pythia v1.0 [28]	VQA [1]	Teney et al. [14] + Deep Layers	72.27 (VQA-v2)	VQA Challenge 2018
Differential Networks [20]	VQA [1], TDIUC [29], COCO-QA [21]	Faster-RCNN [22], Differential Modules [30], GRU [31]	68.59 (VQA-v2), 86.73 (TDIUC), 69.36 (COCO-QA)	AAAI 2019

3.6 基于 Transformer (BERT) 的模型 Transformer Based Models

2018年，BERT腾空出世之后，我们见识到了transformer的强大。所以，最近的工作都是基于transformer的一些模型架设。

ViLBERT

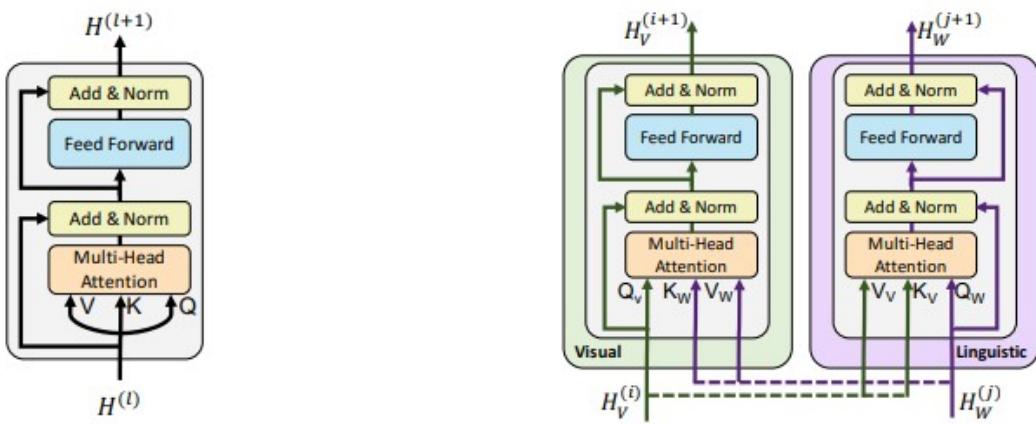
- 双流模型

相关论文：ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks [38]

(Jiasen Lu, 2019) 提出ViLBERT(Vision-and-Language BERT)，该模型学习图像内容和自然语言的无任务偏好的联合表征。

首先，在Conceptual Captions数据集进行Pre-train，然后再迁移到视觉问答，视觉常识推理，指示表达(referring expressions)和基于字幕的图像检索这四个视觉-语言任务。在下游任务中使用ViLBERT时，只需要对基础架构进行略微修改即可。

ViLBERT修改BERT中query条件下的key-value注意力机制，将其发展成一个多模态共注意transformer模块。在多头注意力中交换的key-value对，该结构使得vision-attended语言特征能够融入入视觉表征(反之亦然)。



(a) Standard encoder transformer block

(b) Our co-attention transformer layer

ViLBERT学习的是静态图像及其对应描述文本的联合表征，分别对两种模态进行建模，然后通过一组attention-based的interaction将它们merge在一起。

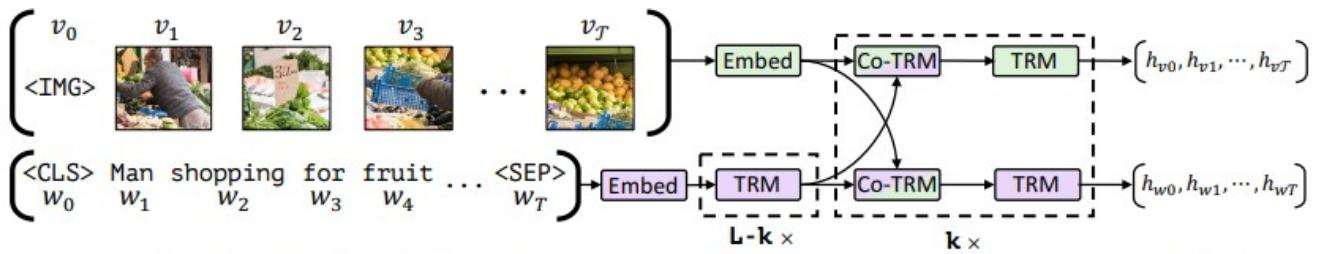
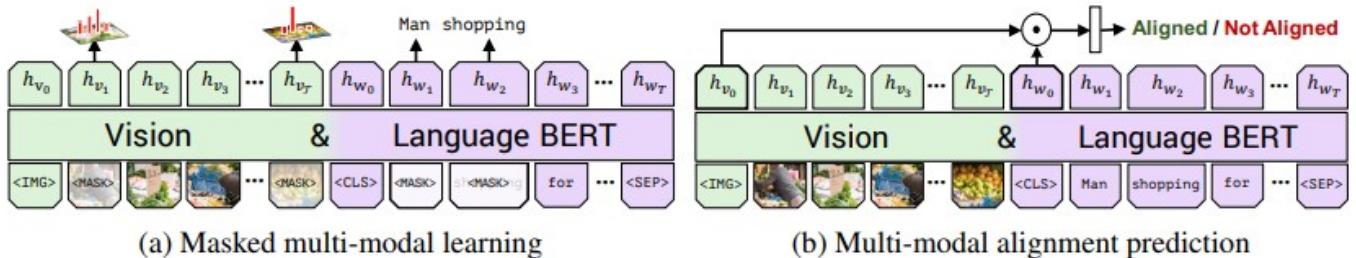


Figure 1: Our ViLBERT model consists of two parallel streams for visual (green) and linguistic (purple) processing that interact through novel co-attentional transformer layers. This structure allows for variable depths for each modality and enables sparse interaction through co-attention. Dashed boxes with multiplier subscripts denote repeated blocks of layers.

该模型由分别作用于图像块和文本段的2个平行BERT模型组成。每个流都是由一系列的transformer blocks和注意力transformer层组成。其attention层是用以两个模态之间特征融合。需要注意的是，流之间的信息交换是被限制于特定层的，所以，文本流在与视觉特征进行交流之前有更多的处理。这也符合我们的直觉，所选择的视觉特征已经相对高级，与句子中的单词相比，视觉特征需要有限的上下文聚合。

预训练任务



训练ViLBERT时采用了2个预训练的任务：

(1) 遮蔽多模态建模

与标准BERT一样，对词和图像输入大约15%进行mask，通过余下的输入序列对mask掉的元素进行预测。对图像进行mask时，0.9的概率是直接遮挡，另外0.1的概率保持不变。文本的mask与bert的一致。vilbert并不直接预测被mask的图像区域特征值，而是预测对应区域在语义类别上的分布，使用pretrain的object-detection模型的输出作为ground-truth，以最小化这两个分布的KL散度为目标。

(2) 预测多模态对齐

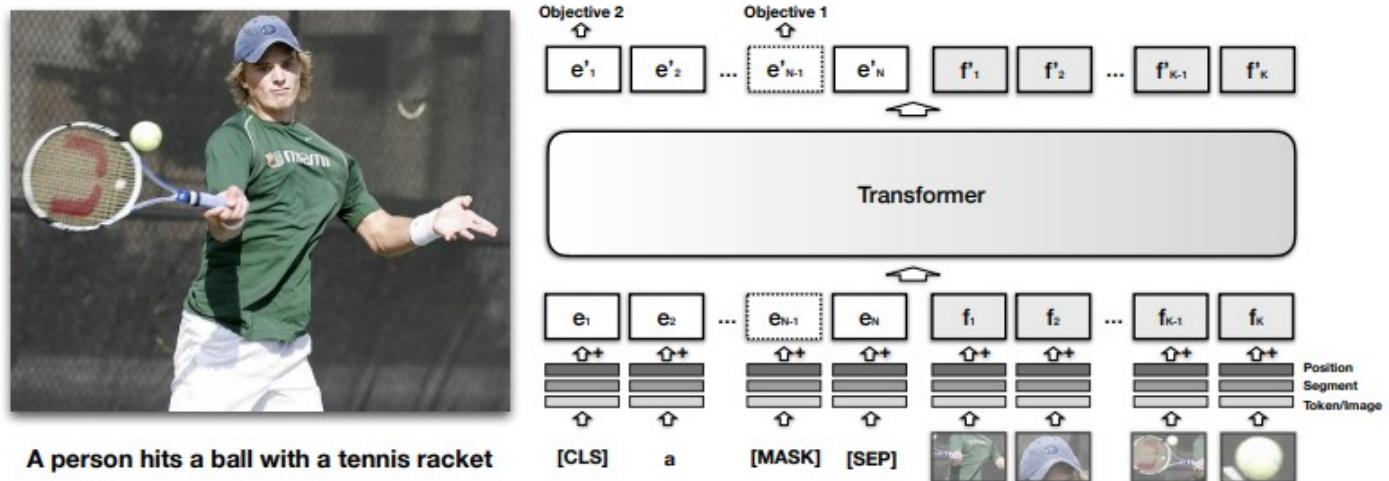
其目标是预测图像-文本对是否匹配对齐，即本文是否正确的描述了图像。以图像特征序列的起始IMG token和文本序列的起始CLS token的输出作为视觉和语言输入的整体表征。借用vision-and-language模型中另一种常见结构，将IMG token的输出和CLS token的输出进行element-wise product作为最终的总体表征。再利用一个线性层预测图像和文本是否匹配。

VisualBERT

- 单流模型

相关论文： VisualBERT: A Simple and Performant Baseline for Vision and Language [39]

(LH Li, 2019) 提出的模型的结构和Video BERT类似，均将text feature和visual feature串联。不同的是，本文的visual feature使用的是region feature，但是没有对其进行mask。



其文字部分的输入为原始的 BERT 文字输入（词向量+位置编码+片段编码）加上 Token/Image 编码来表示其是图片或文字，而图片部分的输入则是采用通过 Faster-RCNN 提取的图片区域特征加上相应的位置编码，片段编码和 Token/Image 编码。

VisualBERT 遵循 BERT 一样的流程，先进行预训练然后在相应的任务上进行微调，其采用了两个预训练任务：第一个是和 BERT 一样的语言掩码，第二个则是句子-图像预测（即判断输入的句子是否为相应图片的描述）。

Github: <https://github.com/uclanlp/visualbert>

LXMERT

- 双流模型

相关论文： LXMERT: Learning Cross-Modality Encoder Representations from Transformers [40]

(H Tan, 2019) 提出了 LXMERT 框架来学习这些语言和视觉的联系，它含有三个编码器：一个对象关系编码器、一个语言编码器和一个跨模态编码器。

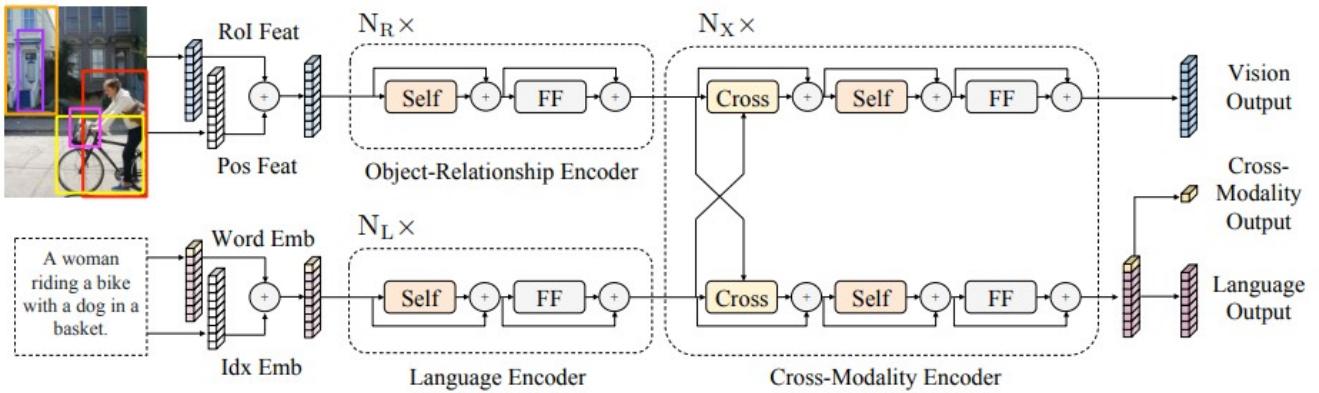


Figure 1: The LXMERT model for learning vision-and-language cross-modality representations. ‘Self’ and ‘Cross’ are abbreviations for self-attention sub-layers and cross-attention sub-layers, respectively. ‘FF’ denotes a feed-forward sub-layer.

Input: 一个图像，相关的句子

通过精心设计和组合这些自我注意和交叉注意层，我们的模型能够生成语言表示、图像表示和跨模式表示。接下来，将详细描述该模型的组件。

- **词级句子嵌入** 一个句子首先被分成 $\{W_1, \dots, W_n\}$, 长度为n, 用的是WordPiece tokenizer。接下来，通过嵌入子层将单词 w_i 及其索引*i* (w_i 在句子中的绝对位置)投影到向量上，然后添加到索引感知词嵌入。
- **对象级图像嵌入** 从图像(上图边框表示)对象检测器检测m个对象 $\{O_1, \dots, O_m\}$ 。每个对象 O_j 由其位置特征 p_j (即，边界框坐标)及其2048维感兴趣区域(Roi)特征 f_j 表示。我们是通过增加2个全连接层的输出来学习位置感知嵌入 v_j

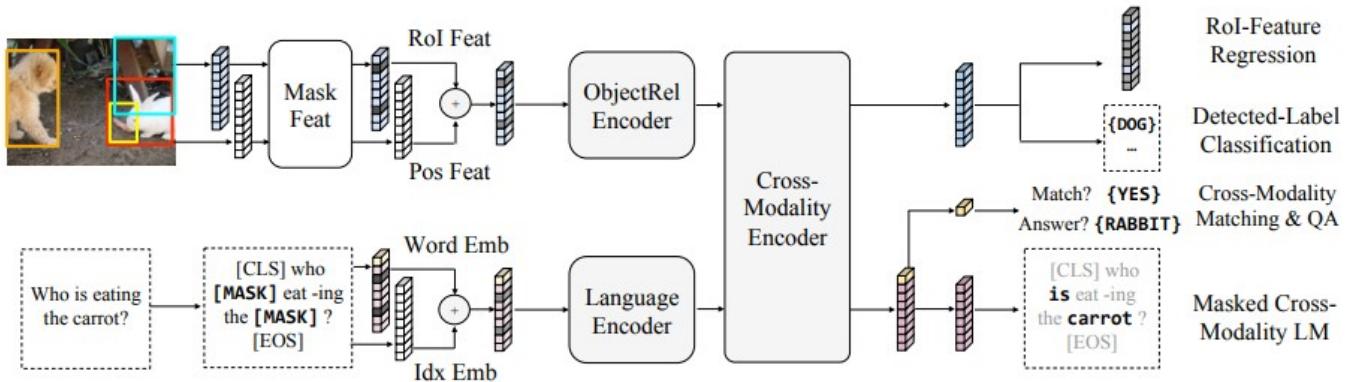
编码器

- **单模态编码器** 在嵌入层之后，他们首先应用两个transformer编码器，即语言编码器和对象关系编码器，它们中的每一个都专注于一个单一的模态（即语言或视觉）。在单模态编码器中（上图中的左侧虚线框），每层都包含一个自注意 (“Self”) 子层和一个前馈 (“FF”) 子层，其中前馈子层进一步由两个完全连接的子层组成。我们分别在语言编码器和对象关系编码器中采用NL层和NR层，在每个子层之后添加一个残差连接和层归一化（在图1中用“+”符号表示）。
- **跨模态编码器** 跨模态编码器中的每个跨模态层（上图中的右侧虚线框）均由两个自注意子层，一个双向交叉注意子层和两个前馈子层组成。在他们的编码器实现中，我们将这些跨模态层叠加N_X层（即，使用第k层的输出作为第 (k + 1) 层的输入）。在第k层内部，首先应用了双向交叉注意子层，该子层包含两个单向交叉注意子层：一个从语言到视觉，一个从视觉到语言。

输出表示

LXMERT跨模态模型有三个输出，分别用于语言、视觉和交叉模态。语言和视觉输出特征序列由交叉模态编码器产生的；对于跨模态输出，他们附加了一个特殊的标记[CLS]（在上图的底部分支中表示为顶部黄色块）在句子词之前，并且该特殊标记在语言特征序列中的对应特征向量为用作交叉模式输出。

预训练任务



- **Language Task: Masked Cross-Modality LM**

任务设置与BERT几乎相同：单词被随机掩蔽，概率为0.15，模型被要求预测这些蒙面词。

LXMERT具有跨模态模型体系结构，可以从视觉模态中预测掩蔽词，从而解决模糊性问题。

- **Vision Task: Masked Object Prediction**

通过随机掩蔽物体（即用零遮掩ROI特征）概率为0.15来预先训练视觉的一面，并要求模型预测这些被掩盖对象的结构。

- **跨模态任务**

为了学习一个强大的交叉模态表示，预先训练LXMERT模型与两个任务，明确需要语言和视觉模式。

- **Cross-Modality Matching** 对于每个句子，概率为0.5，用一个不匹配的句子替换它。然后，训练一个分类器来预测图像和句子是否相互匹配。这个任务有点像在BERT中进行“下一句预测”。
- **Image Question Answering (QA)** 为了扩大训练前的数据集训练前数据中大约1/3的句子是关于图像的问题。当图像和问题匹配时，要求模型预测这些图像相关问题的答案。

Github: <https://github.com/airsplay/lxmert>

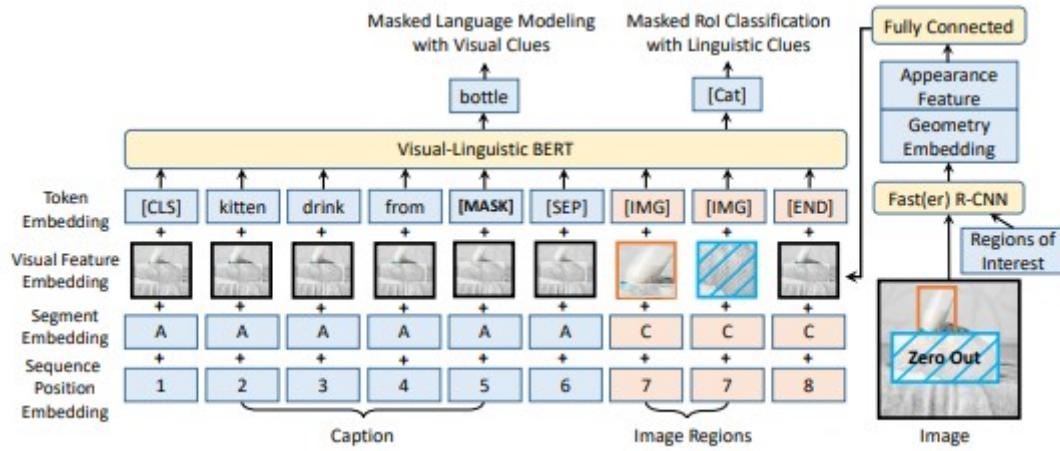
VL-BERT

- **单流模型**

相关论文： VL-BERT: Pre-training of Generic Visual-Linguistic Representations [41]

先前的研究是分别对图像和语言用特定任务的预训练模型进行微调，然后再整合到一起。这样的话，会缺乏一定的“共识”(visual-linguistic)。任务的关键是聚合多模信息(multi-modal information)。为了更好地实现通用表示，作者(W Su, 2019)在大规模的概念标注数据集和纯文本语料库上对VL-BERT进行预训练。

(W Su, 2019)利用transformer处理以视觉与语言嵌入特征的输入。其中输入的每个元素是来自句子中的单词、或图像中的感兴趣区域 (Region of Interests, 简称 RoIs)。获取具有更为丰富的聚合与对齐视觉和语言线索的representation。



Input:

- **Token Embedding** 和BERT是一样的。WordPiece embeddings (Wu et al., 2016) with a 30,000 vocabulary。对于图片，会分配一个[IMG]。
- **Visual Feature Embedding** 每一个输入元素都有一个对应的视觉特征嵌入，视觉特征嵌入是视觉外观特征(visual appearance feature)和视觉几何特征(visual geometry embedding)的concatenation。
- **segment embedding** 定义了三种类型的句段A, B, C，以将来自不同来源的输入元素分开。A表示来自于第一个输入句子中的单词，B表示来自于第二个输入句子中的单词，C表示来自于输入图像的RoI。
- **position embedding** 与BERT一样，将序列位置嵌入添加到每个输入元素中，以指示其在输入序列中的顺序。由于输入的视觉元素之间没有自然顺序，因此它们在输入序列中的任何排列都应获得相同的结果。因此，所有视觉元素的序列位置嵌入都相同。

具体操作：

Step1：图片和文本没法直接对齐，所以暴力输入整张图，直接将图像、文本、segment和position embedding相加作为输入。

Step2：提取图像中的重要部分，增加无文本的图像输入。由于整张图片的粒度远大于文本token，一次性输入整张图片显然不利于图像和文本信息的交互。所以使用了目标检测工具对图片进行分块，提取图像中感兴趣的核心部分RoI (region-of-interest)，加上[IMG]标识，输入到模型中。

1.3.2 预训练任务

• 利用视觉线索(visual clues)对masked语言建模

根据文本+图像信息预测文本token，升级版的MLM。唯一不同的是被mask(15%的概率)的word除了根据没被mask的文本来预测还可以根据视觉的信息来辅助。比如上图中的例子，被mask后的word sequence是kitten drinking from [MASK]，如果没有图片给我们的视觉信息是无法预测出被mask的词是bottle。

• 利用语言线索对masked RoI进行分类

根据文本+图像信息预测RoI的类别，针对图像的“MLM”。图像中的每个RoI都会以15%的概率被随机mask，预训练任务是根据其他线索预测被mask的RoI的类别标签。为了避免由于其他元素的视觉特征嵌入而导致任何视觉线索泄漏，在应用Fast R-CNN之前，将mask RoI中放置的像素置为

0。在预训练期间，对于被mask的RoI的最终输出特征将被feed到具有softmax交叉熵损失的分类器中，以进行对象类别分类。（有个小问题，这里的Fast R-CNN做这个任务的时候，生成的标签感觉不算是ground truth，只能作为辅助训练）

- **图像标题关联预测 (Sentence-Image Relationship Prediction)**

预测图像与标题是否属于同一对。

想法：

VL-BERT使用一个single cross-modal Transformer，让文本和图片信息能更早更多的交互。但是我觉得这种暴力输入图片的时候，会导致输入参数量爆炸，也许不是一个好选择。还有做MLM的时候还是分成了两个任务，有没有办法把这两个任务整合成一个呢？

Github: <https://github.com/jackroos/VL-BERT>

UNITER

- 单流模型

相关论文：UNITER: UNiversal Image-TExt Representation Learning [42]

(Yen-Chun Chen, 2019)介绍了一种通用的图像-文本表示方法UNITER，它是通过对四个图像-文本数据集((COCO, Visual Genome, Conceptual Captions, and SBU Captions)进行大规模的预训练而获得的，可以通过联合多模态嵌入为异构下游V+L任务提供支持。

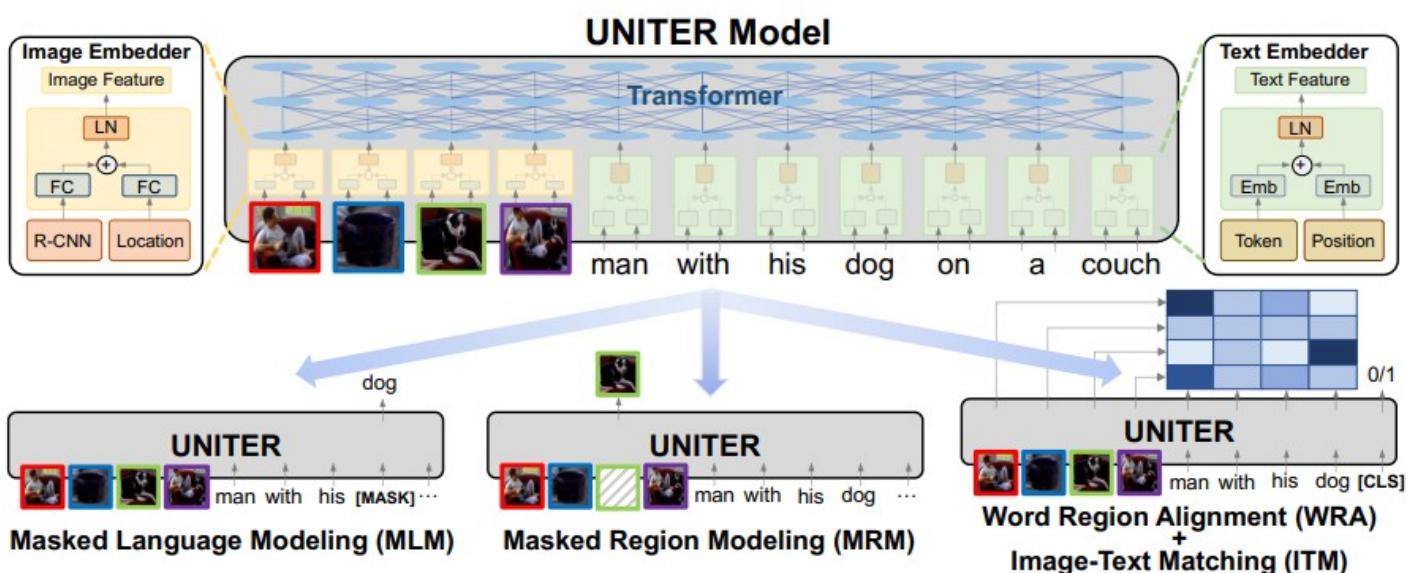


Image Embed:

- Faster R-CNN 提取视觉特征
- 坐标特征编码[上下左右宽高面积]
- 两个特征编码 -> 分别全连接 -> 输出到同一嵌入空间
- 把两个输出加起来 -> Layer Normalization

Text Embed:

- 词嵌入 + 位置嵌入 -> Layer Normalization

预训练任务

- Masked Language Modeling (MLM) (BERT)
- Image-Text Matching (ITM) 图片文字匹配问题
- Word-Region Alignment (WRA)
使用Optimal Transport (自归一, 稀疏化, 搞笑) 来做单词和图片区域匹配的问题
- Masked Region Modeling (MRM)

想法:

涉及到的数据集比较大, 可以做的事情比较多。

ITM的方法不错。

最优传输(Optimal Transport)。

但是感觉没有什么新方法。

Github: <https://github.com/ChenRocks/UNITER>

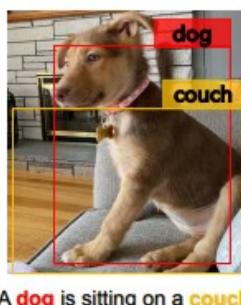
Oscar

大部分来源于 <https://zhuanlan.zhihu.com/p/150261938>

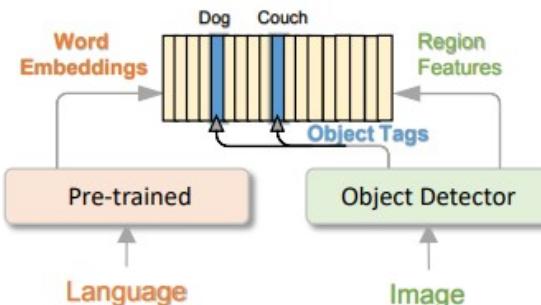
相关论文: Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks [43]

现在基于bert来处理的vision-language task存在的问题: 现在的方式将image region features 和 text features 拼起来, 然后利用自我注意机制以暴力方式学习图像区域和文本之间的语义对齐。

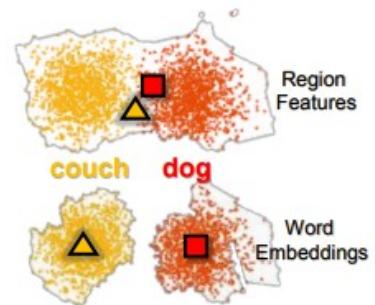
- 由于没有显示的region 与 text poses之间的对齐监督, 因此是一种弱监督的任务。
- 另外, vision region常常过采样(region之间有重叠), 从而带来噪声和歧义 (由于重叠, 导致 region之间的特征区分性不大), 这将会使得vision-language task任务更加具有挑战性。



(a) Image-text pair



(b) Objects as anchor points



(c) Semantics spaces

(Xiajun Li, 2020) 通过引入从images中检测出的object tags 作为anchor points来减轻images 和 text 之间语义对齐的学习。

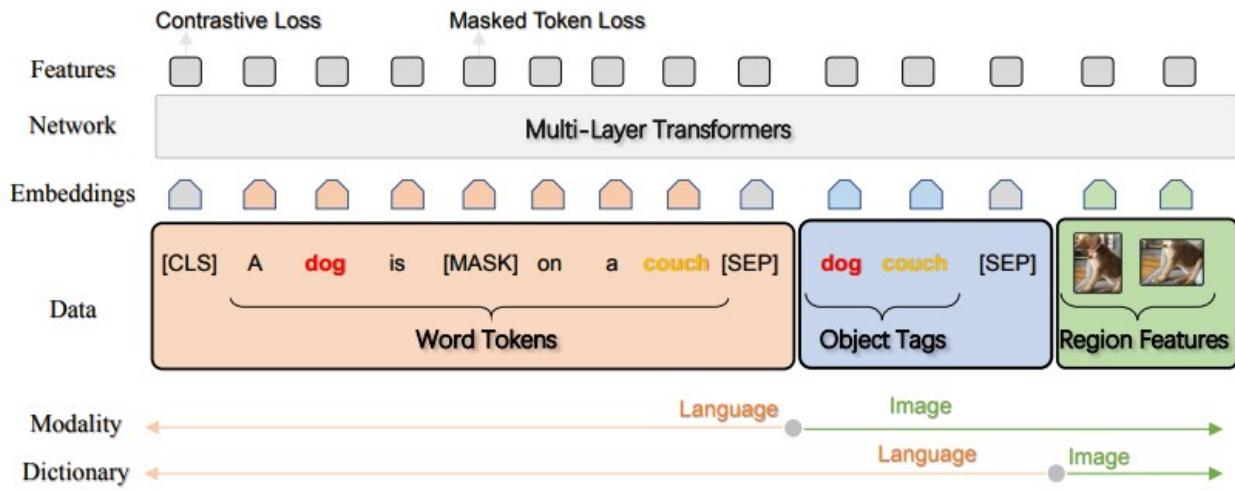


Fig. 3: Illustration of OSCAR. We represent the image-text pair as a triple [word tokens , object tags , region features], where the object tags (e.g., “dog” or “couch”) are proposed to align the cross-domain semantics; when removed, OSCAR reduces to previous VLP methods. The input triple can be understood from two perspectives: a *modality* view and a *dictionary* view.

- 输入表示：将每个(图像-文本)样本定义为一个三元组(单词 序列, 物体标签, 区域特征)。
- 预训练目标：根据三元组中三个项目的分组方式，从两个不同的角度查看输入:模态视角和字典视角。
 - 字典视图的掩码恢复损失，它衡量模型根据上下文恢复丢元素(单词或对象标签)的能力。
 - 模态视角的对比损失，它衡量模型区分原始三元组 及其“污染”版本(即原始物体标签被随机采样的标签替换)的能力。

预训练

1. A Dictionary View: Masked Token Loss, 跟BERT语言掩码一样，对text路和object tags路进行 masked language modeling的训练task。
2. A Modality View: Contrastive Loss, 实际上是一个二分类，当输入的三元组中，原始物体标签 (object tags)被随机替换掉50%，就认为是反例，反之为正例。

想法：

这篇文章如果去掉Object Tags，就跟普通的多模态BERT没有区别了，但这样一个简单的操作，获得了目前Image-base VL-PT文章中所有任务的最好结果。

Github: <https://github.com/microsoft/Oscar>

12-in-1

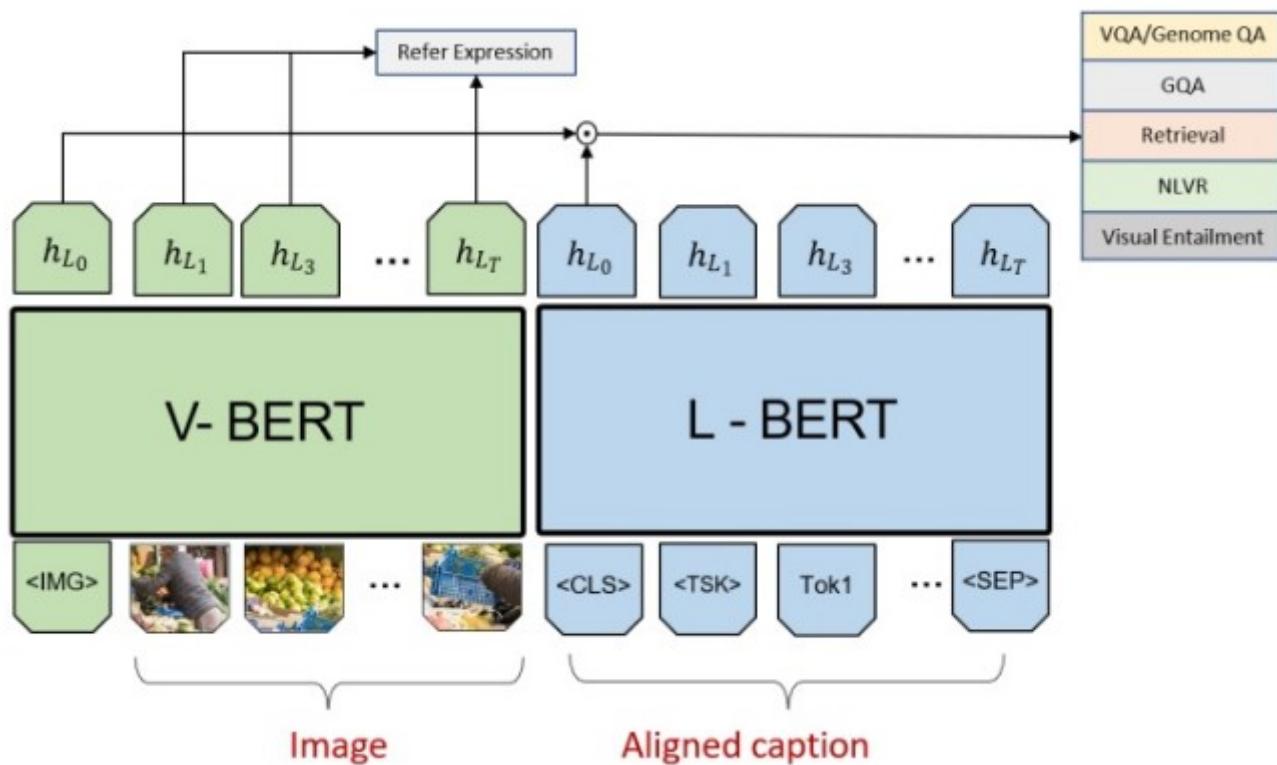
大部分来源于：<https://zhuanlan.zhihu.com/p/150261938>

相关论文：12-in-1: Multi-Task Vision and Language Representation Learning [44]

之前的pre-training工作，都是先进行pre-training，然后在特定的task中进行fine-tune，各个task之间是独立的。这样的话，VQA的模型不能用来做retrieval，每一个新的task都需要去重新fine-tune一个模型。

但其实呢，每个task之间是有存在相互促进的影响的，例如，学着表达「小红花瓶」和理解与回答「小红花瓶是什么颜色的？」是基本相同的概念。联合训练多个任务可以潜在地汇集这些不同的监督任务。而且，单独对于特定task去finetune对于小的数据集容易overfitting，而multi-task同时训练可以减小这个问题。

(J Lu, 2019) 基于ViLBERT开发了一个视觉-语言的多任务模型来学习12个不同的数据集。主要创新点在于设计了一个multi-task的训练策略来联合训练多个task，以及在pre-train task中加入了一些小的trick。



设计了一个multi-task的训练策略，Dynamic stop and go策略。因为在multi-task learning的过程中，有一个比较大的问题，就是有的数据库比较难，有的比较简单，那怎么同时去训练到，因为可能会出现在小数据集上overfitting，而大数据集还训练不够。简单的说，就是它会去监控验证集的结果，就是对于每个task的如果两个epoch过后，他的performance提升少于0.1%，那么就认为他已经收敛了，就停止训练，但如果一直停止的话，多个epoch过后，就可能会丢失掉这个task的信息，所以如果验证集的结果比最好的task减低0.5%之后，在重新开学训练这个task。

Pre-train task

1. Masked multi-modal modelling only for aligned image caption pairs, 跟BERT一样的语言掩码，只是在mask的时候，只在成pair的image-text进行mask，这样可以避免negative samples带来的噪声。
2. Masking overlapped image regions ($IOU > 0.4$)，图片中可能存在一个问题，就是mask掉这个region，去predict它的時候，因为图片存在overlap，可能并不需要依赖文本的信息，只需要通过

这种local信息就能很好的预测出来，如果把这种overlap大0.4的region也mask掉的话，可以强行让网络更多的去依赖文本的信息来进行预测。

Github: <https://github.com/facebookresearch/vilbert-multi-task>

4. 最后

这只是一个简单的整理。

完成于2020.11.1

1. Visual Question Answering using Deep Learning: A Survey and Performance Analysis
URL:<https://arxiv.org/abs/1909.01860> ↵ ↵
2. Visual Question Answering: A Survey of Methods and Datasets
URL:<https://arxiv.org/abs/1607.05910> ↵
3. Survey of Visual Question Answering: Datasets and Techniques
URL:<https://arxiv.org/abs/1705.03865> ↵
4. Github: https://github.com/shengnian/Algorithm_Interview_Notes-Chinese/blob/master/B-自然语言处理/D-视觉问答-1_综述.md ↵
5. Visual Question Answering: Datasets, Algorithms, and Future Challenges URL:
<https://arxiv.org/abs/1610.01465> ↵
6. A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input URL:<https://arxiv.org/abs/1410.0210> ↵
7. Exploring Models and Data for Image Question Answering URL: <https://arxiv.org/abs/1505.02074> ↵
8. VQA: Visual Question Answering URL: <https://arxiv.org/pdf/1505.00468.pdf> ↵ ↵
9. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering URL: <https://arxiv.org/pdf/1612.00837.pdf> ↵
10. Visual Madlibs: Fill in the blank Image Generation and Question Answering URL:
<https://arxiv.org/pdf/1506.00278.pdf> ↵
11. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations URL: <https://arxiv.org/abs/1602.07332> ↵
12. Visual7W: Grounded Question Answering in Images URL: <https://arxiv.org/abs/1511.03416> ↵ ↵
13. CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning URL: <https://arxiv.org/pdf/1612.06890.pdf> ↵
14. TallyQA: Answering Complex Counting Questions URL: <https://arxiv.org/pdf/1612.06890.pdf> ↵
15. TallyQA: KVQA: Knowledge-aware Visual Question Answering URL:
<http://dosa.cds.iisc.ac.in/kvqa/KVQA-AAAI2019.pdf> ↵

16. Answer-Type Prediction for Visual Question Answering <https://www.chriskanan.com/wp-content/uploads/Kafle2016.pdf> ↵
17. A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input URL: <https://arxiv.org/abs/1410.0210> ↵
18. Simple Baseline for Visual Question Answering URL: <https://arxiv.org/abs/1512.02167> ↵
19. URL: <https://arxiv.org/abs/1506.00333> ↵
20. Ask Your Neurons: A Deep Learning Approach to Visual Question Answering URL: <https://arxiv.org/abs/1605.02697> ↵
21. Exploring Models and Data for Image Question Answering URL: <https://arxiv.org/abs/1505.02074> ↵
22. Image Question Answering using Convolutional Neural Network with Dynamic Parameter Prediction URL: <https://arxiv.org/abs/1511.05756> ↵
23. Edge Boxes: Locating Object Proposals from Edges URL: <https://pdollar.github.io/files/papers/ZitnickDollarECCV14edgeBoxes.pdf> ↵
24. Where to look: Focus regions for visual question answering URL: <https://arxiv.org/abs/1511.07394> ↵
25. Stacked Attention Networks for Image Question Answering URL: <https://arxiv.org/abs/1511.02274> ↵
26. Hierarchical Question-Image Co-Attention for Visual Question Answering URL: <https://arxiv.org/abs/1606.00061> ↵
27. Dual Attention Networks for Multimodal Reasoning and Matching URL: <https://arxiv.org/abs/1611.00471> ↵
28. URL: <https://arxiv.org/abs/1708.02711> ↵
29. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering URL: <https://arxiv.org/abs/1707.07998> ↵
30. Pythia v0.1: the Winning Entry to the VQA Challenge 2018 URL: <https://arxiv.org/abs/1807.09956> ↵
31. Focal Visual-Text Attention for Visual Question Answering URL: <https://arxiv.org/abs/1806.01873> ↵
32. Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding URL: <https://arxiv.org/abs/1606.01847> ↵
33. Bilinear CNNs for Fine-grained Visual Recognition URL: <https://arxiv.org/abs/1504.07889> ↵
34. Neural Module Networks URL: <https://arxiv.org/abs/1511.02799> ↵
35. Ask Me Anything: Free-form Visual Question Answering Based on Knowledge from External Sources URL: <https://arxiv.org/abs/1511.06973> ↵
36. Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding URL: <https://arxiv.org/abs/1810.02338> ↵
37. Differential Attention for Visual Question Answering URL: <https://arxiv.org/abs/1804.00298> ↵
38. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks URL: <https://arxiv.org/abs/1908.02265> ↵

39. VisualBERT: A Simple and Performant Baseline for Vision and Language URL:
<https://arxiv.org/abs/1908.03557> ↵
40. LXMERT: Learning Cross-Modality Encoder Representations from Transformers URL:
<https://arxiv.org/abs/1908.07490> ↵
41. VL-BERT: Pre-training of Generic Visual-Linguistic Representations URL:
<https://arxiv.org/abs/1908.08530> ↵
42. UNITER: UNiversal Image-TExt Representation Learning URL: <https://arxiv.org/abs/1909.11740>
↵
43. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks URL:
<https://arxiv.org/abs/2004.06165> ↵
44. 12-in-1: Multi-Task Vision and Language Representation Learning URL:
<https://arxiv.org/abs/1912.02315> ↵