

Video Dialog via Multi-Grained Convolutional Self-Attention Context Networks

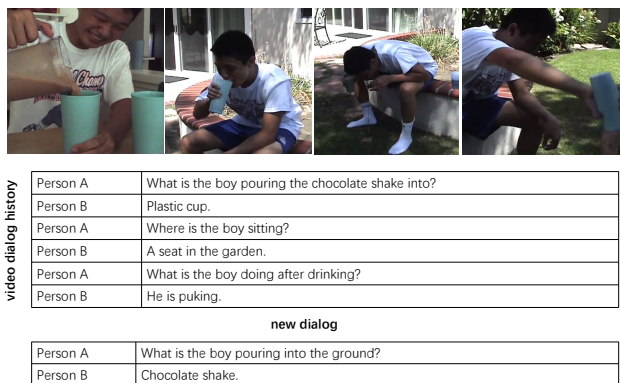


图 1: A simple example of video dialog.

摘要

视频对话是一项新的具有挑战性的任务,需要 AI 代理人以自然语言与人类保持有关视频内容的有意义的对话。具体而言,给定视频,对话历史和关于视频的新问题,代理必须将视频信息与对话历史组合以推断答案。并且由于视频信息的复杂性,图像对话的方法可能无法直接应用于视频对话。在本文中,我们提出了一种新的视频对话方法,称为多粒度卷积自注意上下文网络,它将视频信息与对话历史相结合。我们设计了一种多粒度卷积自注意机制来捕获包含多粒度序列信息的元素和段级别交互,而不是使用 RNN 对序列信息进行编码。然后,我们设计分层对话历史编码器以学习上下文感知问题表示和双流视频编码器以学习上下文感知视频表示。我们在两个大型数据集上评估我们的方法。由于新注意机制的灵活性和并行性,我们的方法可以实现更高的时间效率,广泛的实验也表明了我们的方法的有效性。

1 引言

视觉对话可以看作视觉问题回答 (VQA) 问题的引用,其中需要代理人以自然语言与人类保持关于视觉内容的有意义的对话。与视觉问题回答不同,每个问题都是独立提出的,视觉对话需要代理回答可能与之前对话历史相关的问题。目前,大多数现有的视觉对话方法主要集中在图像。然而,视频也是我们日常生活中常见的视觉信息。因此,我们将图像对话框的任务扩展到视频域,称为视频对话框。具体地,给定视频,对话历史和关于视频内容的新问题,代理必须将视频信息与来自对话历史的上下文信息组合以推断答案。由于视频信息的复杂性,它比图像对话更具挑战性。

由于视频的固有时间结构,当前的图像对话方法可能无法直接应用于缺乏时间建模能力的视频对话。视频问题解答的模型具有这种能力,但是,它们仍然不适合直接利用,因为对话历史上下文的建模不足。对话历史的顺序和相互依赖的属性提出了另一个挑战。如图1所示,为了回答新问题“这个男孩倒在地上是什么?”,需要先前的对话框上下文。如果没有对对话环境的准确理解,系统很难仅通过视觉信息来确定杯子中的液体类型。总之,现有方法的简单扩展难以提供令人满意的结果。因此,需要为视频对话开发新模型。众所周知,递归神经网络 (RNN) 被广泛用于通过循环计算捕获序列信息的能力。然而,基本的 RNN 可能会遇到梯度消失问题,并且很难并行化。一些改进是由几种变体做出的,例如 LSTM,GRU 和 SRU。由于其可并行化的卷积计算,一些工作还采用卷积神经网络 (CNN)。最近,自我关注机制引起了人们对序列建模领域的极大兴趣。它利用来自输入序列的每对元素的注意机制

来生成上下文感知序列表示。因为自我关注的主要操作是矩阵乘法,所以它具有高度可并行性和灵活性,无需任何循环迭代即可进行计算。

在本文中,为了应对视频对话的挑战,我们提出了一种称为多粒度卷积自注意上下文网络的新方法,它将视频信息与对话历史中的上下文信息相结合。一方面,我们发现视频和对话数据的序列长度变化很大,视频长度通常很长。由于这些特性,我们认为在这种情况下 RNN 可能不是最佳选择,由于固有的网络结构,RNN 更适合处理固定长度的信息。另一方面,视频的动态属性(如动作和状态转换)通常包含在几个帧段中而不是单个帧中。然而,最初的自我关注机制仅考虑了元素相互作用。因此,我们不是直接使用一般的 RNN 和自我关注,而是设计了一种多粒度卷积自注意机制,它可以捕获包含多粒度序列信息的元素和段级交互。通过使用这种新的注意机制,我们设计了一个分层对话历史编码器来学习上下文感知问题表示和一个双流视频编码器来学习上下文感知视频表示。然后,我们融合视觉和文本信息,以便在新一轮对话中回答问题。我们将整个网络命名为 MGCSACN。本文的主要贡献如下:

- 与以往主要关注静态图像的研究不同,我们将图像对话的任务扩展到视频领域,称为视频对话,这更具挑战性。我们提出了一种称为多粒度卷积自注意上下文网络的新方法,以学习视觉和文本信息的联合表示。
- 我们开发了一种多粒度卷积自我关注机制,可以捕获元素和段级别的交互,其中包含多粒度的序列信息。我们在视觉和文本序列编码过程中利用这种新颖的注意机制,其中通常应用 RNN。
- 我们的方法在两个大型数据集上实现了最先进的性能。并且它的时间成本也比其他基于 RNN 的方法更有效,这得益于我们完全自我关注和基于 CNN 的网络结构的灵活性和并行性。

2 相关工作

视觉对话可以看作是视觉问答 (VQA) 的扩展,它将单回答问题扩展到多回转对话。因此,在本节中,我们首先介绍视觉问答的一些相关工作。然后,我们回顾一些现有的视觉对话方法。

Visual Question Answering. 对于图像问答, Malinowski 等人提出了早期工作。它通过贝叶斯方法将语义分析与图像分割相结合。由于深度学习在计算机视觉和自然语言处理方面都表现出很高的效果,因此提出了许多基于神经网络的方法。朱等人将空间注意力添加到标准 LSTM 模型中。Yu 等人开发语义注意机制来选择与问题相关的高级概念。Nguyen 等人提出了一个密集的共同注意网络,它在输入图像和问题之间采用密集的对称交互来改善特征融合。与图像问答相比,视频问答的探索相对较少。Jang et al. 提出了一种基于双 LSTM 的方法,同时兼顾空间和时间的关注。Na et al. 开发了一个读写存储器网络,用于融合多模态特征。% 并通过多级卷积神经网络存储时间信息。最近, Gao et al. 提出了一个共存记忆网络来捕捉运动和外观信息的进一步交互。Liang et al. 提出了一种新的神经网络,叫做 Focal Visual-Text Attention network (FVTA), 用于视觉问答的集体推理。

Visual Dialog. 视觉对话可以看作是视觉问答 (VQA) 的扩展,它将单回答问题扩展到多回转对话。具体地,每一轮中的问答配对可以参考来自对话的先前上下文的信息。正如中提出的,视觉对话需要根据图像和对话历史来预测给定问题的答案。虽然对话系统已被广泛探索,但视觉对话仍然是一项年轻的任务。直到最近,还提出了一些不同的方法。例如, Das 等人分别提出了基于后期融合,基于注意力的分层 LSTM 和存储器网络的三种模型。他们还通过将 Amazon Mechanical Turk 上的两个主题配对来聊聊图像(提出更好地想象隐藏图像的问题)来提出 VisDial 数据集。Lu 等人提出了一种发生器鉴别器结构,其中使用来自预训练鉴别器的感知损耗来改进发生器的输出。De Vries 等人提出了一个 GuessWhat 游戏风格数据集,其中一个人询问有关图像的问题以猜测哪个对象已被选中,第二个人回答问题。Das et al. 建议使用深度强化学习来学习

“Questioner-Bot”和“Answerer-Bot”的策略,基于选择两个代理正在谈论的正确图像的目标。Seo 等人基于具有注意记忆的新注意机制解决对话问题中的视觉引用,其中模型通过注意力检索过程间接地解析表达的共同参照。Kottur et al . 提出了关于视觉共鸣的内省推理,它明确地将共识联系起来并将它们置于图像中的单词级别,而不是隐含地或在句子级别,如在先前的视觉对话工作中。Massiceti et al . 提出了 FLIPDIAL, 一种用于视觉对话的生成卷积模型,它能够生成答案,并根据视觉上下文生成问题和答案。Lee et al . 提出了“问答者心中的回答”(AQM), 这是一个使用信息理论方法的实用的面向目标的对话框,其中,提问者通过选择合理的方式来确定回答者的意图通过明确计算候选意图的信息增益和每个问题的可能答案来提出问题。吴等人提出了一个顺序共同注意生成模型,它可以共同推理图像,对话历史与问题,以及一个可以动态访问注意记忆的鉴别器,并获得中间奖励,并实现了 VisDial 数据集的最新技术。

上述工作主要集中在图像对话框上。至于视频对话的任务,它仍然没有被探索过。赵等人提出了类似的工作。他们通过采用分层注意上下文学习方法和循环感知问题理解的循环神经网络以及学习联合嵌入视频表示的多流注意网络来研究多转视频问题回答的问题。他们还提出了两个来自 YouTubeClips 和 TACoS-MultiLevel 的大型多转视频问答数据集。最近,Hori 等人提出了一个模型,该模型将基于多模态注意力的视频描述技术整合到端到端的对话系统中。与这些工作不同,我们利用更有效和高效的卷积自注意上下文网络进行视频对话任务。

3 PROPOSED APPROACH

3.1 Problem Formulation

对于视频对话仍然没有被探索,这里我们首先介绍一些基本概念和术语。我们在 V 中用 \mathbf{v} 表示视频,用 C 表示 \mathbf{c} 的对话历史记录,在 Q 中用 \mathbf{q} 表示新问题,相应的答案由 \mathbf{a} 分别为 A 。对于视频是一系列静态帧,视频 \mathbf{v} 的帧级表示由 $\mathbf{v}^f = (v_1^f, v_2^f, \dots, v_{T_1}^f)$, 其中 T_1 是视频 \mathbf{v}

中的帧数。视频 \mathbf{v} 的细分级表示由 $\mathbf{v}^s = (v_1^s, v_2^s, \dots, v_{T_2}^s)$, 其中 T_2 是段数, v_j^s 是 j -th 段的表示通过预先训练的 3D-ConvNet。然后 C 中的对话历史 $\mathbf{c} = (c_1, c_2, \dots, c_N)$ 给出, 其中 c_i 是 i -th 圆形对话, 其中包含问题 q_i 并回答 a_i 。使用这些符号, 视频对话的任务可以表述如下。给定一组视频 V 和相关的对话历史 C , 视频对话任务的目标是训练一个模型, 该模型在询问有关视觉内容的新问题时学习生成类似人的答案。与视频问题回答任务类似, 有两种类型的模型可以产生答案, 生成和判别。对于生成解码器, 采用字序列发生器 (通常是 RNN) 来拟合地面实况答案序列。对于判别解码器, 提供了另外的候选答案词汇表, 并且该问题被重新表述为多类别分类问题。

3.2 多粒度卷积自注意上下文网络

在本节中, 我们将介绍用于视频对话的多粒度卷积自注意上下文网络, 它遵循编码器 - 解码器网络结构。如图2所示, 整个模型由对话历史编码器, 上下文感知视频编码器和应答解码器组成。在下文中, 我们将分别描述它们的细节。

3.2.1 Multi-Grained Convolutional Self-Attention. 在介绍主编码器 - 解码器部分之前, 我们首先描述我们提出的多粒度卷积自注意 (MGCSA) 机制, 用于从输入中捕获多粒度交互信息序列。它用于我们的对话历史和视频信息编码过程。类似于先前的工作我们采用正弦 - 余弦位置编码机制将序列的时间信息添加到自我关注过程中, 为了简单起见, 我们将在以下描述中忽略它。如图2(c) 所示, MGCSA 单元接受一系列字嵌入或视频帧特征作为输入, 表示为 $X = (x_1, x_2, \dots, x_N)$ 。首先, 我们将输入序列拆分为等长 l 的 k 段, 由 $X = (X^1, X^2, \dots, X^k)$ 给出, 其中 $X^1 = (x_1, x_2, \dots, x_l)$, $X^2 = (x_{l+1}, x_{l+2}, \dots, x_{2l})$, \dots 和 $n = k$ 次。为了直观理解, 我们以不同的灰度显示不同的片段。如果输入序列不能被平分, 则序列将被填充以满足条件。然后, 我们将自我关注机制应用于

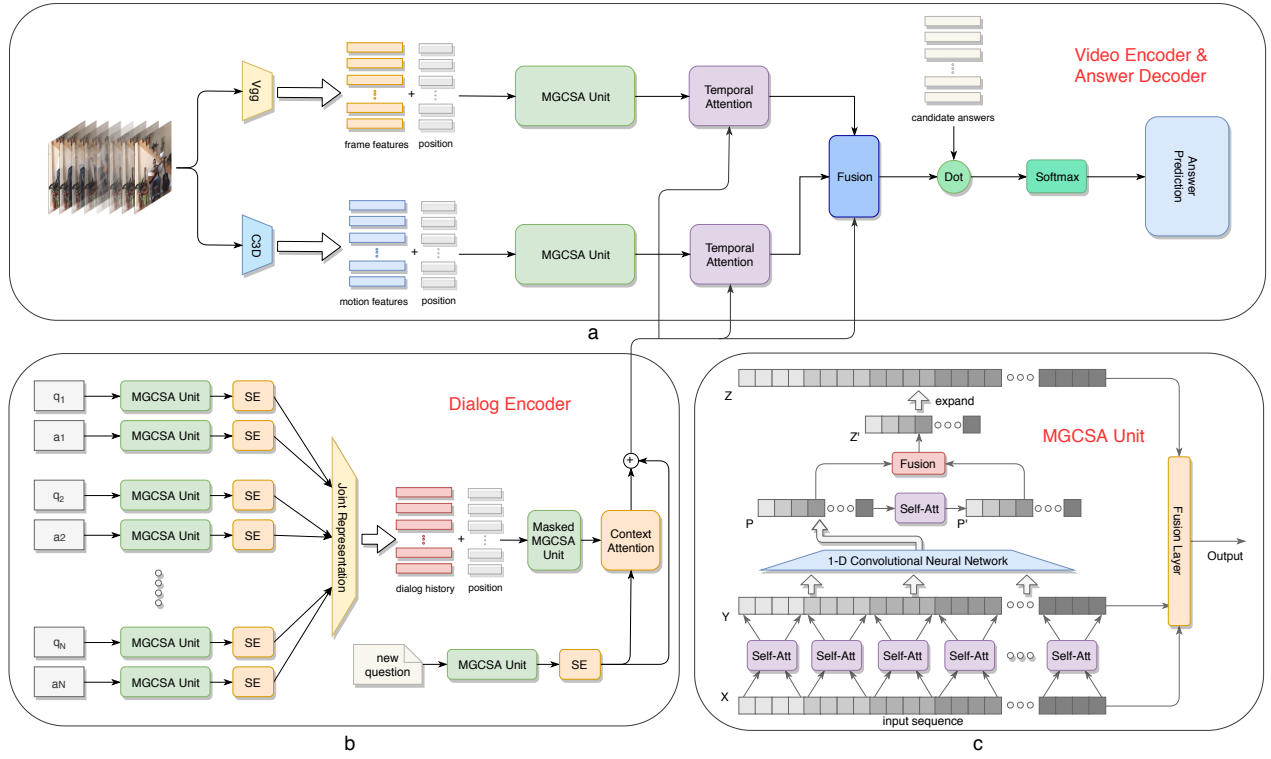


图 2: The Overview of Multi-Grained Convolutional Self-Attention Context Networks for Video Dialog. It consists of three parts. (a) 视频编码器和应答解码器, 其中我们首先学习基于多粒度卷积自我关注和时间注意机制的情境感知联合视频表示然后融合多模态信息以预测最终答案。(b) 对话编码器, 其中我们利用相同的注意机制对对话历史和新问题进行编码。(c) MGCSA 单元, 其中我们详细显示了多粒度卷积自我关注机制。

每个段, 以便捕获每个段内的本地交互信息, 由

$$Y^i = \text{Attention}(X^i, X^i, X^i), i = 1, 2, \dots, k \quad (1)$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \quad (2)$$

其中 d 是序列元素的维度, Y^i 是自我关注后的新的 i -th 段表示。现在我们获得一个新的序列 $Y = (Y^1, Y^2, \dots, Y^k)$ 乘法, b_g^1, b_g^2 是偏向量和 S, S' 是门的得分向量, 其值在 0 和 1 之间。给定新序列 $Z' = (z'_1, z'_2, \dots, z'_k)$, 我们为 l 复制每个 z'_i 以获得一个新序列 $Z = (z_1, z_2, \dots, z_N)$ 最后, 我们使用相同的特征融合操作来组合输入序列 X , 元素级上下文序列 Y 和段级上下文序列 Z , 并生成最终的多粒度上下文感知序列表示 R , 由

列 Z' 由

$$S = \sigma(W_g^1[P; P'] + b_g^1), \quad (3)$$

$$S' = \sigma(W_g^2[P; P'] + b_g^2), \quad (4)$$

$$Z' = S' \odot P' + S \odot P \quad (5)$$

$$F_{yz} = \text{Fusion}(Y, Z), \quad (6)$$

$$R = \text{Fusion}(F_{yz}, X) \quad (7)$$

其中 $Fusion()$ 包括相同的作为方程 (3,4,5) 的操作, 具有不同的权重参数。

3.2.2 对话历史编码器. 在本节中, 我们将详细介绍分层对话历史编码器, 它使用对话历史来学习连贯的问题表示。如图2(b)所示, 给定对话历史 $\mathbf{c} = (c_1, c_2, \dots, c_N)$ where c_i 是 i -th 圆形对话, 包含问题 q_i 和答案 a_i , 我们首先使用多粒度卷积自我关注 (MGCSA) 来学习 q_i 的问题表示和 i 回合中的答案 a_i 。因为 MGCSA 单元的输出仍然是一系列字嵌入, 我们使用类似于的句子嵌入 (SE) 机制来获得句子表示。以下等式用于计算句子嵌入的输出。

$$\begin{aligned} f(x_i) &= \text{softmax}(W_1 \tanh(W_2 x_i + b_1)), \quad (8) \\ O &= \sum_{i=1}^n f(x_i) \odot x_i \quad (9) \end{aligned}$$

其中 W_1, W_2 是权重参数, b_1 是偏见, n 是输入序列的长度, x_i 是输入的一个元素。对于 i 轮对话历史记录, 我们获得问题 r_i^q 的句子级别表示, 并在句子嵌入后回答 r_i^a 。然后, 我们在这个问题 - 答案对上执行联合表示机制来学习 i -th 圆形对话历史 c_i 的表示, 由

$$c_i = \tanh(W_c^1 r_i^q + W_c^2 r_i^a) \quad (10)$$

其中 $W_c^1 \in \mathbb{R}^{d_c \times d}$ 和 $W_c^2 \in \mathbb{R}^{d_c \times d}$ 是用于问答表示融合的投影矩阵。 d 是 r_i^q 和 r_i^a 的维度, 而 d_c 是联合表示的维度。 \tanh 是元素双曲正切函数, 在多模态表示融合中表现良好。与单转问题回答不同, 视频对话的上下文是相关的。因此, 我们利用掩蔽的多粒度卷积自注意来编码上下文交互。这里, 掩码用于避免当前轮次的对话看到后一轮对话, 这符合我们的常识。基于联合问答表示的交互式对话框上下文表示 $\mathbf{c} = (c_1, c_2, \dots, c_N)$ 由 $\mathbf{u} = (u_1, u_2, \dots, u_N)$, 其中 $u_i \in \mathbb{R}^{d_c}$ 。

接下来, 给出一个新问题, 使用与前一轮对话相同的过程来学习问题表示 q 。我们使用它来通过注意机制进一步过滤掉与新问题相关的上下文信息。注意得分 s_i^{qu} 由

$$s_i^{qu} = w_{qu}^T \tanh(W_{qu}^1 q + W_{qu}^2 u_i + b_{qu}) \quad (11)$$

其中 $W_{qu}^1 \in \mathbb{R}^{d_m \times d}$, $W_{qu}^2 \in \mathbb{R}^{d_m \times d_m}$ 是参数矩阵和 $w_{qu}^T \in \mathbb{R}^{d_m}$ 是参数向量。 $b_{qu} \in \mathbb{R}^{d_m}$ 是偏向量, d_m 是中间维度。然

后我们应用 softmax 函数来生成对话框上下文的注意力分布, 由

$$\alpha_i^{qu} = \frac{\exp(s_i^{qu})}{\sum_i^N \exp(s_i^{qu})} \quad (12)$$

因此, 参与的对话框上下文表示由 $u^q = \sum_i^N \alpha_i^{qu} U_i$ 。最后的上下文感知问题表示由

$$q^u = q + u^q \quad (13)$$

3.2.3 Context-Aware 视频编码器. 在本节中, 我们描述了视频编码器的一部分, 它学习用于答案预测的上下文感知视频表示。我们知道视频包含丰富的视觉信息, 例如外观, 物体, 动作等。很多工作已经证明有必要将这些信息考虑在内以获得高质量的视频理解。在这里, 我们考虑外观和运动信息来捕获帧级和分段级视频表示。具体来说, 我们利用预先训练的 VGGNe 来提取外观特征, 并使用 3D-ConvNet 来获取运动特征。如部分 3.1所示, 外观序列为 $\mathbf{v}^f = (v_1^f, v_2^f, \dots, v_{T_1}^f)$ 和动作序列是 $\mathbf{v}^s = (v_1^s, v_2^s, \dots, v_{T_2}^s)$ 。

首先, 我们还利用所提出的多粒度卷积自注意机制来学习外观信息和运动信息的多粒度表示, 表示为 $\mathbf{v}'^f = (v_1'^f, v_2'^f, \dots, v_{T_1}'^f)$ 和 $\mathbf{v}'^s = (v_1'^s, v_2'^s, \dots, v_{T_2}'^s)$, 另外。然后, 给定上下文感知问题表示 q^u , 我们开发了一个临时关注过程, 根据问题和对话历史上下文, 使用目标信息本地化相关的帧或段, 因为视频丰富的冗余信息。对于 i -th 帧 $v_i'^f$, 其临时关注分数 s_i^{qf} 由

$$s_i^{qf} = w_{qf}^T \tanh(W_{qf}^1 q^u + W_{qf}^2 v_i'^f + b_{qf}) \quad (14)$$

其中 $W_{qf}^1 \in \mathbb{R}^{d_n \times d}$, $W_{qf}^2 \in \mathbb{R}^{d_n \times d_f}$ 是参数矩阵, 而 $w_{qf}^T \in \mathbb{R}^{d_n}$ 是参数向量。 $b_{qf} \in \mathbb{R}^{d_n}$ 是偏向量。 d_n 是中间维度, d_f 是外观特征维度。 Softmax 函数仍然用于生成视频帧上的注意力分布, 由

$$\alpha_i^{qf} = \frac{\exp(s_i^{qf})}{\sum_i^{T_1} \exp(s_i^{qf})} \quad (15)$$

因此, 上下文感知视频外观表示由

$$v^{qf} = \sum_i^{T_1} \alpha_i^{qf} v_i'^f \quad (16)$$

另一方面, 相同的操作应用于视频运动信息。我们可以获得上下文感知视频运动表示 v^{qs} 。因此, 最终的上下文感知视频表示由

$$v_q^{fs} = v^{qf} \odot v^{qs} \quad (17)$$

, 其中 \odot 是元素产品运营商。

最后, 我们将上下文感知视频表示 v_q^{fs} 与上下文感知问题表示 q^u , 用于以下问题预测, 由

$$f_{quv} = \text{Concat}(g(v_q^{fs}), g(q^u)) \quad (18)$$

其中 $\text{Concat}(\cdot)$ 是一个连接两个输入向量的函数 $g(\cdot)$ 是门控双曲正切激活。

3.2.4 答案解码器. 在这里, 我们根据现有的视觉问题回答模型, 将视频对话问题建模为具有预定义候选答案集的分类任务。鉴于上一节中定义的最终上下文感知视频 - 文本融合表示 f_{quv} , 我们现在为视频对话任务开发一种多项选择方法。我们首先通过采用另一个多粒度卷积自注意单元来学习答案集中每个候选答案的语义表示 a_i^c 。然后, 我们可以得到答案表示矩阵 $A = [a_1^c; a_2^c; \dots; a_{T3}^c] \in \mathbb{R}^{T3 \times d_h}$ 用于答案集, 其中 $T3$ 是候选答案的数量, 答案表示的维度 d_h 是与最终融合表示 f_{quv} 相同。最后, 我们计算最终融合表示 $f_{quv} \in \mathbb{R}^{d_h}$ 和每个候选答案之间的相似性, 并使用 softmax 函数将可能的答案分类为

$$p_a = \text{softmax}(A \times f_{quv}^T) \quad (19)$$

其中 $p_a \in \mathbb{R}^{T3}$ 是候选人的概率分布答案。必须注意的是, 视频对话也可以看作是一项生成任务, 将在我们未来的工作中讨论。例如, 代替使用 softmax 函数进行答案分类, 也可以通过将最终融合表示 f_{quv} 作为输入, 利用 LSTM 答案生成器生成自由格式的自然语言答案。

4 实验

4.1 实验构建

4.1.1 数据集. 我们在这里使用的数据集是在中提出的, 它们是从 YouTubeClips 和 TACoS-MultiLevel 中提取

的。这两个数据集分别包含 1,987 和 1,303 个视频。每个 YouTubeClips 视频由 60 帧组成, 每个 TACoS-MultiLevel 视频由 80 帧组成。在这两个数据集中, 每个视频都有五个不同的对话框, 由五对来自专业公司的众包工人生成。YouTubeClips 数据集中有 6515 个视频对话框, TACoS-MultiLevel 数据集中有 9935 个视频对话框。相应地, YouTubeClips 数据集和 TACoS-MultiLevel 数据集中对话框问答对的数量分别为 66,806 和 37,228。统计上, 在 TACoS-MultiLevel 数据集中的大多数视频对话框中有五轮对话, 并且 YouTubeClips 数据集中每个视频对话框的问答对数量大多在三到十二之间。根据构建的视频对话框的数量, 两个数据集的训练数据, 验证数据和测试数据的百分比分别为 90%, 5% 和 5%。

此外, 每个视频对话的地面实况答案与基于欧几里德距离的所有其他答案之间的语义相似性与预先训练的手套嵌入计算, 以将前 50 个答案排名为每个视频的候选答案对话。

4.1.2 评估指标. 我们选择三个通用的评估标准 MRR, P @ K 和 MeanRank 来评估所提出的 MGCSACN 方法和其他基线模型的性能。给出 Q 中的新问题 q , 其真实答案为 a^t , 我们用 r_{a^t} 表示问题 q 的真实答案的等级 q 。我们现在介绍下面的评估标准。

- **MRR.** MRR 测量地面实况答案的排名质量, 由

$$MRR = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{r_{a^t}^q},$$

其中 $|Q|$ 是使用的测试问题的数量。

- **P @ K.** P @ K 衡量排名靠前的答案的排名精确度。P @ K 度量由

$$P@K = \frac{\sum_{q \in Q} 1[r_{a^t}^q \leq K]}{|Q|},$$

其中 $1[\cdot]$ 是指标函数。

- **MeanRank** MeanRank 测量地面实况答案的平均等级位置, 由

$$\text{MeanRank} = \frac{\sum_{q \in Q} r_{a^t}^q}{|Q|}$$

4.1.3 实现. 构建视频对话系统的预处理过程如下所示。首先, 我们使用预先训练的 word2vec model 来获得对

话框的语义表示。单词向量的维度为 100, 总词汇表的大小为 6,500。其次, 我们将每个帧的大小调整为 224 美元/次 224 美元, 并使用预先训练的 VGGNet 来获取每个帧的特征表示。同时, 预训练的 3D-ConvNet 提取视频的 4,096 维运动特征表示。具体地, 每个运动段包含 16 个帧并且与两个数据集的相邻段重叠 8 个帧。

在 MGCSA 单元中, 段的长度是一个重要的超参数, 它影响段级序列交互。我们将视频序列的片段长度设置为 5, 将问题和答案设置为 3。实际上, 根据不同种类的输入序列, 段长度的选择是动态的。不正确的段长度会影响性能并占用冗余内存。一方面, 段长度太小可能会降低段级交互的影响, 这使得它与元素级交互没有区别。另一方面, 段长度太大可能影响段级序列信息的多样性。我们做了很多实验来获得最终设置。MGCSA 单元的输出维数和对话历史中问答对的联合表示的维度也是两个必要参数, 并且进行了一些实验以获得最优条件。具体来说, 我们将 MGCSA 维度从 128 更改为 1024, 将联合表示维度从 64 更改为 512。因此, 我们将 MGCSA 维度和联合表示维度都设置为 128。

至于训练过程, 我们将交叉最小化通过使用 Adam optimizer 来减少萎缩, 其中初始学习率设置为 0.0005, 指数衰减率设置为 0.8。另外, 我们应用梯度裁剪方法来控制 1.0 内的梯度范数, 以防止在反向传播过程中出现大的梯度。

4.2 性能比较

由于视频对话是一项新任务, 我们将一些现有的图像对话和视频问答模型 (类似于) 扩展为视频对话的基线模型, 介绍如下:

- **ESA method** 只使用注意机制生成给定问题和视频特征的融合表示, 而不使用对话历史。
- **ESA +** 方法通过添加分层 LSTM 网络来对对话历史进行建模, 从而扩展了 ESA 模型。在这种方法中, 我们将给定问题与对话历史融合以构造更复杂的联合表示。

- **STVQA +** 方法通过添加分层 LSTM 网络来扩展 STVQA 模, 以模拟对话历史并使用双 LSTM 网络融合对话历史和视频功能。
- **STAN +** 方法通过添加分层 LSTM 网络来扩展 STAN 模型, 以模拟对话历史并利用时空关注对话历史。
- **CDMN +** 方法通过添加分层 LSTM 网络来扩展 CDMN 模型来建模对话历史, 并使用运动外观共存网络来同时学习运动和外观特征。
- **LF +, HRE + 和 MN +** 方法通过利用 LSTM 网络对视频信息进行编码来扩展三个模型, 视频信息分别基于后期融合, 基于注意力的分层 LSTM 和存储器网络。
- **SFQIH +** 方法通过使用 LSTM 网络对视频信息进行编码来扩展 SF-QIH-se 模型, 该视频信息连接每个可能的答案选项的所有输入嵌入并使用相似性网络预测可能答案的概率分布。

除基线模型外, 我们还将我们的方法与最新工作进行比较。目前, 关于视频对话的现有工作很少。赵等人提出了一项类似的工作, 叫做多转视频问答。他们使用 LSTM 和注意机制来编码对话历史和给定问题以获得联合问题表示。然后, 它通过多流注意网络将这种联合表示与视频特征相结合。并采用多步推理策略来提高推理能力。这个模型有一些变种。没有帧信道分层时空关注上下文网络的方法由 HACRN_(w/o.f) 标记, 没有运动信道时间关注上下文网络的方法由 HACRN_(.m)。此外, 没有多步推理过程的方法由 HACRN_(w/o.r) 表示, 并且具有多步推理过程的方法由 HACRN_(r) 表示。

与上述工作不同, 我们的方法 (MGCSACN) 使用多粒度卷积自我关注机制来编码视觉和文本信息, 而不是直接使用一般 RNN 和注意力, 而是捕获多粒度序列信息。表 1 显示了 TACoS-MultiLevel 数据集的实验结果, 表 2 演示了 YoutubeClip 数据集上的实验结果。通过分析这些实验结果, 我们得出了几个有趣的结论:

- 包含扩展视频 QA 模型的结果表的第一部分显示了视频信息对视频对话的影响。对视频内

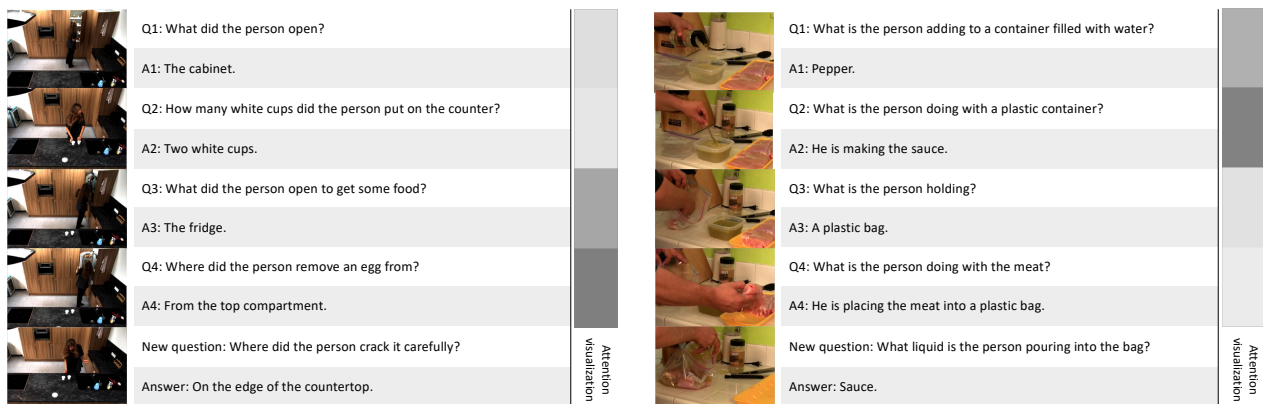


图 3: The visualization of dialog context attention.

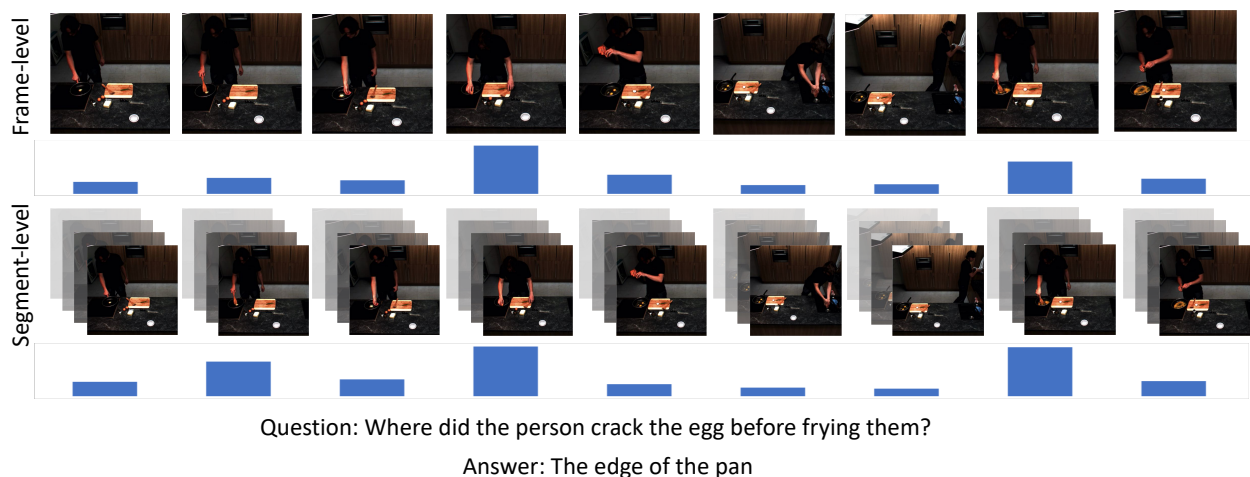


图 4: The visualization of video attention in both frame level and segment level.


容有更好理解的模型可以在视频对话任务中获得更好的性能。

- 包含扩展图像对话框模型的结果表的第二部分显示了视频对话框的历史对话框信息的重要性。更好地理解对话框上下文的模型在视频对话任务中表现更好。
- 我们的方法 MGCSACN 几乎在所有标准中都比所有基线模型和最新的 HACRN 模型表现更好。这一事实表明了我们充分关注和基于 CNN 的网络结构的有效性,该网络结构采用多粒度的

卷积自注意机制,可以捕获来自双流视频表示和对话历史的元素和段级别交互。

4.3 定性分析

为了直观地理解我们的模型,我们在方法中分析了部分注意机制。在图 3 中,我们展示了对话框编码器中对话框上下文关注的可视化,其中问答对根据它们与问题的相关性得到各种分数。如图 3 的左侧部分所示,第 4 轮问答案对引起了更多关注,指出新问题中的“它”是“蛋”。在正确的部分,它是第二轮引起更多关注。此外,图 4 显示了帧级和段级视频注意的可视化。正如



Conversation Context	Question	Answer
Q1: Where were the two persons? A1: On the playground. Q2: Who kicked a football? A2: The man in red. Q3: Where was the man in white? A3: He was behind the camera. Q4: What knocked down the camera? A4: The football.	What was the man in white doing before he dodged the football?	MN+: He was standing on the ground. SFQIH+: He was jumping away. STVQA+: He was jumping away STAN+: He was staring at the ground. CDMN+: He was laughing. HACRN: He was standing behind the camera. MGCSACN: He was taking photos. Ground Truth: He was taking photos.

图 5: An example of video dialog experimental results on YoutubeClip dataset.

表 1: Experimental results on TACoS-MultiLevel dataset.

Method	MRR	P@1	P@5	MeanRank
ESA	0.411	0.298	0.515	11.964
ESA+	0.411	0.300	0.507	10.435
STVQA+	0.427	0.305	0.540	9.762
STAN+	0.452	0.319	0.594	8.401
CDMN+	0.454	0.317	0.597	8.376
LF+	0.434	0.281	0.625	6.438
HRE+	0.454	0.313	0.594	8.813
MN+	0.467	0.343	0.625	7.688
SFQIH+	0.468	0.375	0.656	6.313
HACRN _(w/o.m)	0.444	0.319	0.579	8.726
HACRN _(w/o.f)	0.452	0.324	0.583	8.622
HACRN _(w/o.r)	0.512	0.391	0.643	6.625
HACRN _(r)	0.526	0.386	0.682	5.804
MGCSACN	0.542	0.437	0.717	5.875

我们所看到的，问题中有两个重要的动作，即破蛋和煎炸。因此，片段级注意力的直方图显示出对包括油炸黄油，开裂蛋和煎蛋的片段的高度关注。显然，它包含干扰信息。因此，我们需要更详细的帧级信息。框架

表 2: Experimental results on YoutubeClip dataset.

Method	MRR	P@1	P@5	MeanRank
ESA	0.333	0.224	0.418	11.571
ESA+	0.396	0.252	0.541	8.412
STVQA+	0.411	0.266	0.578	7.284
STAN+	0.418	0.274	0.577	7.258
CDMN+	0.422	0.278	0.584	7.074
LF+	0.389	0.250	0.563	8.531
HRE+	0.413	0.281	0.594	7.688
MN+	0.422	0.313	0.563	8.594
SFQIH+	0.441	0.313	0.656	6.781
HACRN _(w/o.m)	0.443	0.283	0.635	6.149
HACRN _(w/o.f)	0.454	0.295	0.636	6.042
HACRN _(w/o.r)	0.469	0.315	0.661	5.792
HACRN _(r)	0.470	0.306	0.670	5.496
MGCSACN	0.481	0.344	0.687	6.969

级别的注意力成功地集中在与蛋相对应的框架上。结合这两种注意机制，视频对话系统可以根据问题更好地理解视频。此外，YoutubeClip 数据集的实验结果示例如图 5 所示。与基线模型相比，我们的模型在此示例中表现更好，可以得到准确的答案。

表 3: Time cost of different methods.

Method	Training Time(s)	Inference Time(s)
HACRN _(w/o.r)	0.519	0.166
HACRN _{GRU}	0.485	0.169
NSACN	0.151	0.050
MGCSACN	0.153	0.089

4.4 时间分析

在本节中，我们将方法的时间效率与某些基于 RNN 的模型进行比较。为了有效地控制变量，我们保持所有类似的超参数相同，例如批量大小，特征维度。此外，应该保证将给定问题与对话历史和视频特征联合融合的过程保持一致。对于 HACRN 模型，我们选择没有多步推理机制的变体来避免其影响。我们还用 HACRN_{GRU} 记录的 GRU 单元替换 HACRN 模型的 LSTM 单元。NSACN 模型将 MGCSACN 的 MGCSA 单元替换为正常的自注意结构，这种结构相对简单，只考虑元素级交互。所有这些型号都在相同的硬件环境 (E5-2678 v3, 1080ti GPU 和 128GB 内存) 中进行评估。

实验结果如表 3 所示。每批的推理时间仅包含前向传播的时间，每批的训练时间包括反向传播和前向传播。与 HACRN_(w/o.r) 相比，我们的方法在训练时间减少了 70.5%，在推理时间减少了 46.4%。HACRN_{GRU} 比 HACRN_(w/o.r) 更快，但是，它仍然比我们的方法慢得多。虽然我们的方法不如 NSACN 快，但我们方法的答案预测性能要好得多，如表 4 和 5 所示。这些事实表明我们的方法的高时间效率。

4.5 消融研究

我们进行消融研究以评估每个组件的贡献并分析我们模型中的最佳设计选项。首先，我们删除对话历史并检查性能以说明其重要性。具体来说，我们使用常规问题表示来替换上下文感知问题表示。同样，我们分别删除视频的外观特征和动作特征，以证明不同视频信息的重要性。TACoS-MultiLevel 数据集的实验结果

表 4: Ablation study results on TACoS-MultiLevel dataset.

Method	MRR	P@1	P@5	MeanRank
MGCSACN _(o.v)	0.473	0.344	0.625	7.531
MGCSACN _(o.a)	0.469	0.344	0.594	6.281
MGCSACN _(o.m)	0.501	0.313	0.656	7.438
NSACN	0.499	0.366	0.650	6.950
MGCSACN	0.542	0.437	0.717	5.875

表 5: Ablation study results on YoutubeClip dataset.

Method	MRR	P@1	P@5	MeanRank
MGCSACN _(o.v)	0.365	0.250	0.500	9.125
MGCSACN _(o.a)	0.404	0.281	0.531	9.375
MGCSACN _(o.m)	0.416	0.313	0.563	8.781
NSACN	0.459	0.312	0.593	7.437
MGCSACN	0.481	0.344	0.687	6.969

列于表 4 中，YoutubeClip 数据集的结果显示在表 5 中，其中 MGCSACN_(o.v), MGCSACN_(o.m), MGCSACN_(o.a) 表示没有相应参与对话历史，外观特征和运动特征的模型。正如我们从这些结果中看到的那样，完整模型 MGCSACN 比所有 MGCSACN_(o.v), MGCSACN_(o.m) 和 MGCSACN_(o.a) models 证明视频的对话历史，外观和动作信息的有效性。并且 MGCSACN 模型优于 NSACN 模型的事实表明了我们的多粒度卷积自注意机制的有效性。

5 结束

在本文中，我们将图像对话的任务扩展到视频域，称为视频对话。为了解决这个问题，我们提出了一种称为多粒度卷积自注意上下文网络的新方法，它将注意机制与卷积神经网络相结合，对视频和对话环境进行编码。具体来说，我们设计了一种多粒度卷积自我关注机制，它可以捕获元素和段级别的交互，其中包含

多粒度序列信息。通过使用这种新的注意机制，我们采用分层对话历史编码器来学习上下文感知问题表示，并使用双流视频编码器来学习视频表示。然后，我们融合视觉和文本信息，为新问题生成相应的答案。它是一个完全自我关注和基于 CNN 的模型，以摆脱一般 RNN 网络的限制。由于网络结构的灵活性和并行性，我们的方法可以实现高时间效率。基于两个大型数据集的广泛实验也显示了我们方法的有效性。在未来的工作中，我们将探索更具生成性的视频对话系统。

6 竞彩分析

1. 目前，大多数现有的视觉对话方法主要集中在图像而不是视频上，这也是一种视觉信息的表现形式，它渗透到我们的日常生活中，并包含更复杂的时空内容。视频对话任务中心在对话历史的连续性和相互依存性方面面临的一个重大挑战，在以往的研究中尚未处理。因此，视觉对话更具挑战性，能够帮助机器更好地理解视觉和文本内容。2. 提出的多粒度卷积自注意机制不仅对时间成本更为有效，而且能够捕获元素和段级信息。3. MGCSACN 可以有效地得到多粒度序列表示和多模态联合表示。并且，这与前人的方法不大一样。1. 提出了一种新的方法，即多粒度卷积自注意上下文网络来学习视觉和文本信息的联合表示，而不是简单地处理静态图像或与上下文无关的问题。2. 不再使用 RNN 来获取序列信息，而是开发了一种多粒度卷积自注意机制，这种机制具有高度的并行性和灵活性，可以在不进行任何重复迭代的情况下进行计算。3. 在视频对话任务中，采用了识别模型和生成模型来预测答案。有效生成形式与识别形式生成的答案相比，不仅在语言上具有灵活性，而且能够掌握视觉和文本语境中的重要信息和逻辑依赖性，它是由词汇词典而不是有限的候选集生成的。

7 可以拓展进步的空间

关于预测答案，有两种类型的模型，即区分模型和生成模型。在识别模型中，译码器应该从一个给定的候选集中选择地面真值应答。因此，可以将其视为一个

分类问题。另一方面，生成模型必须从字典逐字生成准确自然的答案。由于生成类人答案的必要性，生成模型预测的答案不仅要符合视觉和文本语境，还要符合语法。这比简单计算最终特征表示的相似性的识别性更具挑战性。有效生成形式与识别形式生成的答案相比，是从词汇词典中生成的答案，而不是从有限的候选集中生成的答案，它不仅语言流畅，而且能够掌握视觉和文本上下文中的重要信息和逻辑依赖性。

不妨如下定义新的问题对于生成形式，解码器预测生成的答案 $\mathbf{a}_g \in A_g$ ，其中 \mathbf{a}_g 中的单词包含在字典 D 中。然后利用一个 transformer 结构对其进行 target sequence 信息对 multi-modal 式 fusion 过程，因为 video representation 分为 appear 与 motion 的，所以用两条 distribution stream 进行 fusion。然后做两个 ablation study 对其分析。首先是两条 video stream 的意义，其中将两个 video 串接进行 attention。另外一个 ablation study 则将 decoder 替换成 RNN 版本的，分析性能与时间消耗。