SUTD-TrafficQA: A Question Answering Benchmark and an Efficient Network for Video Reasoning over Traffic Events

Li Xu He Huang Jun Liu*
Information Systems Technology and Design
Singapore University of Technology and Design
{li_xu, he_huang}@mymail.sutd.edu.sg, jun_liu@sutd.edu.sg

Abstract

Traffic event cognition and reasoning in videos is an important task that has a wide range of applications in intelligent transportation, assisted driving, and autonomous vehicles. In this paper, we create a novel dataset, SUTD-TrafficQA (Traffic Question Answering), which takes the form of video QA based on the collected 10,080 in-the-wild videos and annotated 62,535 QA pairs, for benchmarking the cognitive capability of causal inference and event understanding models in complex traffic scenarios. Specifically, we propose 6 challenging reasoning tasks corresponding to various traffic scenarios, so as to evaluate the reasoning capability over different kinds of complex yet practical traffic events. Moreover, we propose ***Eclipse****, a novel* ***Eff****i****c****ient g****li****m****pse*** *network via dynamic inference, in order to achieve computation-efficient and reliable video reasoning. The experiments show that our method achieves superior performance while reducing the computation cost significantly. The project page: https://github.com/SUTDCV/SUTD-TrafficQA.*

1. Introduction

Intelligent transportation [64] has been receiving increasing attention recently, and for the applications, such as assisted driving, violation detection, and congestion forecasting, accurate and efficient cognition and reasoning over the traffic events captured by video cameras is extremely important. As shown by previous works [2, 18], well-designed datasets are often crucial for the development, adaptation and evaluation of different data-driven approaches. This indicates the significance of creating comprehensive and challenging benchmarks for video causal reasoning and cognitive development of models, that explore the underlying causal structures of various traffic events. To this end, we introduce a novel dataset, SUTD-TrafficQA (Traffic Question Answering), to facilitate the research of causal reasoning in complex traffic scenarios.

In our dataset, to help develop models for addressing several major and concerning issues in intelligent transportation, we design 6 challenging reasoning tasks, which require exploring the complex causal structures within the inference process of the traffic events. As shown in Figure 1, these tasks correspond to various traffic scenarios involving both road-agents and surroundings, and the models are required to forecast future events, infer past situations, explain accident causes, provide preventive advice, and so on.

To present these reasoning tasks, video question answering [66] is a natural and effective choice, and is used for our dataset construction, since to accurately answer the given questions, the models need to acquire strong capabilities of performing various levels of logical reasoning and spatio-temporal cognition for the events.

Besides providing the challenging and useful reasoning tasks, we adopt a combination scheme of online collection and offline capturing to collect videos, such that the data in our benchmark covers various traffic events, diversified road-agents and surroundings, and different capturing perspectives in the wild. With the provided various tasks and diverse videos, our dataset shall be able to serve as a comprehensive benchmark for video reasoning of traffic events.

In some application scenarios, (e.g., assisted driving), the computational resource and energy budget can be constrained. Thus both the inference accuracy and the computation efficiency are important for video event reasoning in these scenarios. Existing video QA methods [23, 10, 30] mainly focus on strengthening the reasoning accuracy without emphasizing much efficiency, and most of these works apply fixed computation pipelines to answer different questions, while ignoring to conduct adaptive and efficient computation resource allocation based on the logic structure behind reasoning over video events.

In this paper, to achieve reliable and efficient video reasoning, we propose **Eclipse**, an **Eff**i**c**ient g**li**m**pse** network. Specifically, considering there is often large redundancy

*Corresponding Author.

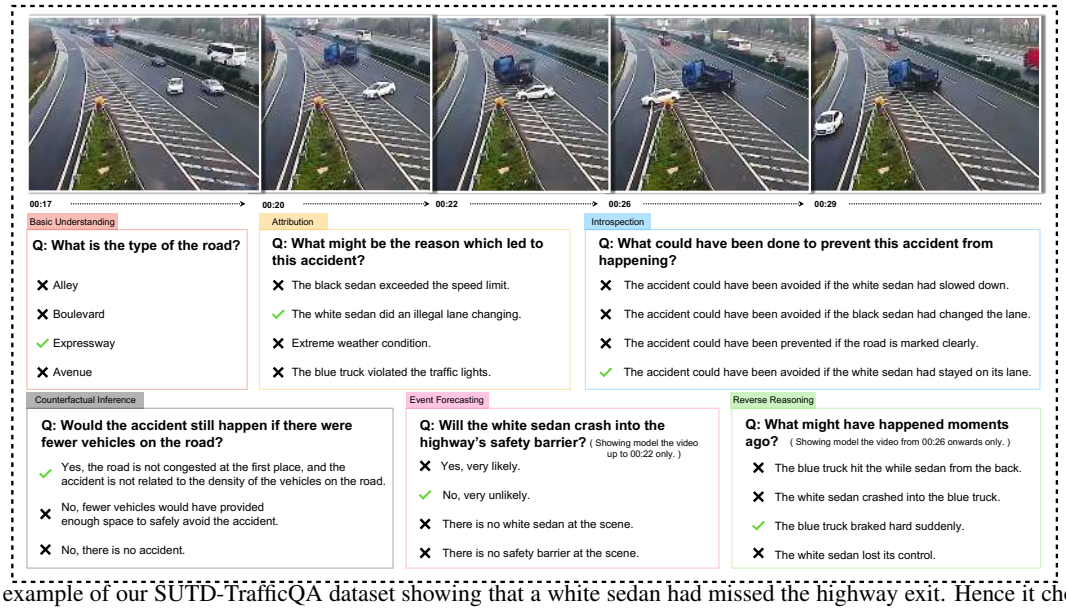


Figure 1. An example of our SUTD-TrafficQA dataset showing that a white sedan had missed the highway exit. Hence it chose to change the lane illegally for driving towards the exit. To avoid collision, the blue truck had to brake suddenly, and then an accident occurred. Six reasoning tasks are designed, covering a broad range of inference problems from basic understanding to complex reasoning and attribution analysis. To accurately answer these questions, the models need to explore the causal, logic, and spatio-temporal structures of the video event.

among video frames, via dynamic inference, our network adaptively determines where to skip and glimpse at each step, and what computation granularity needs to be allocated for the glimpsed frame. Such a dynamic reasoning scheme avoids feature extraction for the irrelevant segments in the video, and hence significantly reduces the overall computation cost towards reliable and efficient reasoning. It is noteworthy that both the determination of selecting a glimpse frame and the decision of computation granularity for each glimpse are essentially discrete operations, which are not trivial to optimize. To handle this issue, an effective joint Gumbel-Softmax mechanism is also introduced in this paper, which makes our Eclipse framework fully differentiable and end-to-end trainable.

To the best of our knowledge, this is the first work that simultaneously performs adaptive frame localization and feature granularity determination in a novel dynamic reasoning process for reliable and efficient causal reasoning and video QA. A joint Gumbel-Softmax operation is also introduced in this work to optimize the two decisions jointly.

2. Related Works

Intelligent Transportation. With the rapid development of deep learning techniques [28, 39, 49], data-driven intelligent transportation [5, 33, 58, 5] has emerged as a prominent research topic. Lou et al. [33] presented a dataset together with an adversarial learning model for vehicle re-identification. Different from existing intelligent transportation datasets and methods, in this paper, we investigate the problem of causal reasoning with video QA over various traffic scenarios. A new benchmark, SUTD-TrafficQA, together with a novel model, Eclipse, is proposed for this challenging task.

Video QA Datasets. Recently, there emerges a great interest in visual reasoning and question answering in videos [40, 34, 66, 25], and several video QA datasets [54, 63, 47, 13, 12, 17, 43, 61] have been developed. Among them, MovieQA [47] and TVQA [29] present the movie and TV-show videos respectively with human-generated questions. More recently, CLEVRER [56] focuses on collision event reasoning among several simple visual objects in a controlled environment using fully synthetic videos. Differently, our SUTD-TrafficQA focuses on reasoning over the complex traffic scenarios in the wild, where 6 challenging tasks for traffic event reasoning are introduced based on the diverse real-world traffic videos.

Video QA Methods. Extensive studies have been conducted for video QA [60, 12, 16, 22, 23, 30, 48, 11, 24, 59, 45, 50, 8, 32, 62, 65]. Yu et al. [60] employed LSTM to encode videos and QA pairs, and adopted an attention mechanism [57]. Jang et al. [12] used LSTMs with a different attention scheme to capture spatio-temporal patterns in videos. Different from existing video QA methods, our Eclipse model investigates the direction of learning an effective glimpse policy for adaptive reasoning to achieve reliable reasoning with computation efficiency.

Computation-Efficient Models. Recent works [3, 35, 44, 14, 27, 53, 42, 52] have pointed out the need of improv-

Table 1. Comparison among SUTD-TrafficQA and some other video QA datasets. Providing challenging **traffic-scenario reasoning tasks** with **real-world videos** and **human-generated QA pairs**, our dataset shall serve as a comprehensive and challenging benchmark for video reasoning over traffic events.

Dataset	Synthetic Videos	Real-World Videos						
	CLEVRER [56]	MovieQA [47]	MSRVTT-QA [54]	TGIF-QA [12]	TVQA [29]	MarioQA [36]	Social-IQ [13]	**SUTD-TrafficQA (Ours)**
Basic Understanding	✓	✓	✓	✓	✓	✓	✓	✓
Attribution	✓	✓	×	×	✓	✓	✓	✓
Event Forecasting	✓	×	×	×	×	×	×	✓
Reverse Reasoning	×	×	×	×	×	×	×	✓
Counterfactual Inference	✓	×	×	×	×	×	×	✓
Introspection	×	×	×	×	×	×	×	✓
QA Generation	Automatic	Human	Automatic	Automatic & Human	Human	Automatic	Human	Human
Topic	Synthetic Object Collision	Movies	Various Scene	GIFs	TV-Shows	Gameplay	Social Behavior	Traffic Events

ing the computation-efficiency when designing deep models, and different strategies, including filter pruning [31], weight sparsification [46], vector quantization [1], and dynamic routing [51], etc., have been proposed. In this paper, we propose an efficient model, Eclipse, the first model that performs dynamic inference and adaptive computation adjustment for video QA, which leverages an effective glimpse policy with the guidance of text and visual context information for both glimpse frame selection and computation granularity determination.

3. SUTD-TrafficQA Dataset

Our dataset contains 62,535 QA pairs and 10,080 videos of traffic scenes. Below we first propose 6 challenging traffic-related reasoning tasks, and then introduce the QA collection process and the dataset statistics.

Basic understanding. This task evaluates the ability of the models in perceiving and understanding traffic scenarios at the basic level, which consists of multiple sub-tasks including feature-query (e.g., vehicle type, road situation, and environment description), event-query (e.g., accident existence, pedestrian action analysis, and events temporal relation), event classification (e.g., accident type), and counting (e.g., road-agent number).

Event forecasting. This task requires a model to infer future events based on observed videos, and the forecasting questions query about the outcome of the current situation.

Reverse reasoning. This task is to ask about the events that have happened before the start of a video segment.

Counterfactual inference. This task queries the consequent outcomes of certain hypothesis (e.g., what if the blue sedan had not accelerated?). The hypothetical conditions do not occur in the video, so the model needs to reason about the imagined events under the designated condition.

Introspection. This task is to test if models are able to provide preventive advice (e.g., what could the pedestrian have done to avoid the collision with the car?). The candidate answers list actions that could have been taken to avoid traffic accidents or congestion.

Attribution. This task seeks the explanation about the causes of traffic events (e.g., what are the reasons of the rear-end crash?), so as to check if models are able to infer the underlying factors leading to the event.

We define all the above reasoning tasks as multiple-choice questions without limiting the number of candidate answers for each question. The number of candidate answers varies from 2 to 12 for different questions. We then sample among the candidate answers to balance the dataset and limit the occurrence of the same correct answers within each task to minimize language biases. As summarized in Table 1, by introducing these challenging tasks, our dataset complements existing datasets, and facilitates the exploration of video QA in complex traffic scenarios.

3.1. QA Collection

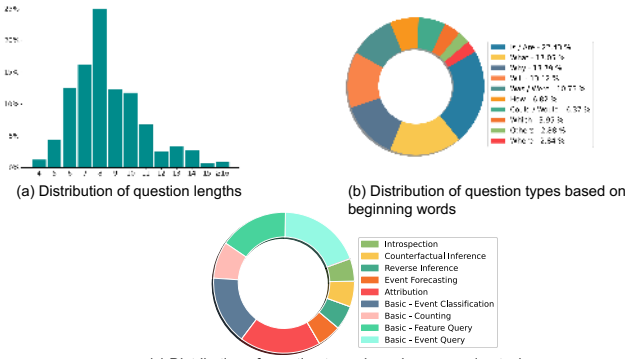
Videos. We collected videos by using a combination of online harvesting and offline capturing, to cover various real-world traffic scenarios. As for online video collection, a variety of video sharing platforms, based in different countries, are used to increase the diversity, including but not limited to YouTube, LiveLeak, Twitter, and Bilibili. More than nine thousand videos were thus collected from these online sources. As for offline video capturing, a set of videos were captured via handheld cameras by volunteers, while another set were fetched from car-mounted video recorders. These two sets of offline videos were then examined and trimmed into around one thousand video clips.

After combing the online videos and offline captured ones, we obtain a total of $10,080$ videos containing diversities in various aspects, including: a) different weathers (sunny/rainy/windy/snowy); b) different time (daytime/night); c) diverse road situations (congested/sparse, urban/rural roads); d) various traffic events (accidents, vehicle turning, pedestrian behaviors, traffic lights, etc.); e) different video perspectives (surveillance camera perspective/car-mounted video perspective/hand-held camera perspective); and f) various clip lengths (from 1 to 70 seconds).

QA pairs. As the first step, the 6 tasks were explained to annotators. To ensure they fully understand the design principle of the tasks, multiple example questions were prepared for them to identify which task the questions belong to. Afterwards, each annotator was presented with batches

of video folders, where each folder contains 100 clips that were randomly selected from the full video set. Annotators were asked to create at least 3 questions of the reasoning tasks for each video. We did not impose constraints on question formats to encourage annotators to keep their QA pairs diversified. Specifically, we encourage them to rephrase similar questions or candidate answers to push the models to learn underlying semantics of QA pairs rather than superficial language correlation. To ensure the quality of QA pairs, we cross-checked QA annotations on a weekly basis. In addition, we kept monitoring the distribution of different tasks in QA collection to maintain the balance and diversity of our dataset.

3.2. Dataset Statistics



(a) Distribution of question lengths

(b) Distribution of question types based on beginning words

(c) Distribution of question types based on reasoning tasks

Figure 2. Statistics of SUTD-TrafficQA. More in supplementary.

In this part, we present the statistics of our SUTD-TrafficQA dataset. Figure 2 (a) demonstrates the distribution of question length measured by number of words. The average length of questions is 8.6 words. Figure 2 (b) shows the various question types categorized by their beginning words, which implies the diversity of questions in our dataset. Figure 2 (c) presents the split of questions in terms of reasoning tasks. Our dataset covers a broad range of traffic-related reasoning tasks requiring various levels of spatio-temporal understanding and causal reasoning in videos.

4. Eclipse Network

To deal with video reasoning, a common solution is to watch the full video and analyze the whole event information carefully. In this manner, generally, a fixed computation architecture [29, 12, 21, 10] can be applied over the whole video for tackling each question. However, using a fixed network architecture for handling the video QA task is often computation-heavy and energy-consuming [44, 42], because the video sequence used for reasoning can be very long and contain plenty of frames for processing.

Recalling that, as humans, to analyze events in a video, we may not be patient enough to scrutinize the frames in the whole video, instead, we may adopt an adaptive information "foraging" strategy [15, 7]. Concretely, we may use a "dynamic" inference manner to skip forth and back over the sequence to progressively infer and select some useful frames based on the task. Moreover, for the picked frames, we may examine a few of them very carefully while glancing over others. Such a dynamic and adaptive perception habit [15, 7] frees us from watching the whole video thoroughly, and often enables fast yet still very accurate video reasoning at a very small frame usage.

Motivated by this, we aim to explore the direction of efficient and dynamic reasoning in complex traffic scenarios. Thus, we propose an **Effi**cient g**lim**pse (**Eclipse**) network for video QA, as illustrated in Figure 3. Instead of using a fixed computation architecture over each video and question, our network learns to dynamically skip to and select a useful video frame at each inference step. Moreover, our network adaptively decides the feature computation granularity (i.e., coarse or fine) of the selected frame. To perform such a process, at each inference step, our network takes advantage of the guidance information including the QA pair, the currently selected frame and the historical cues for dynamic reasoning.

Specifically, as shown in Figure 3, in our network, to provide QA information for the Interaction Module, the QA Bank stores the representation of the QA pairs. At each inference step, to assist the dynamic reasoning of selecting the frame and the corresponding feature granularity, the Interaction Module leverages the QA information, the currently selected frame and the information from historically observed frames to derive an expressive representation, which then serves as the input of the dynamic reasoning process performed by the downstream modules, including the Prediction Module for outputting the reasoning result, and Glimpse-Determination Module for dynamically determining which frame to be observed at next step. Besides, the Exit-Policy Module also uses this representation to adaptively decide whether we can exit the reasoning process at current inference step. Via such a dynamic and recurring reasoning process, our network can derive reliable answer predictions with notable computation efficiency w.r.t. both the frame usage and feature computation granularity. We elaborate the network modules in detail below.

QA Bank. To provide the QA information as the guidance for dynamic reasoning, our QA Bank encodes the representation of the question via a bi-directional LSTM. By concatenating hidden states of the BiLSTM, the question representation can be denoted as $H^q \in \mathbb{R}^{n_q \times 2d}$, where n_q is the number of words in the question, d denotes the dimension of LSTM hidden state. Similarly, we encode all the candidate answers as $\{H^{a_i}\}_{i=1}^{N}$, where $H^{a_i} \in \mathbb{R}^{n_{a_i} \times 2d}$, N is the number of candidate answers, and n_{a_i} is the number of words in the answer a_i.

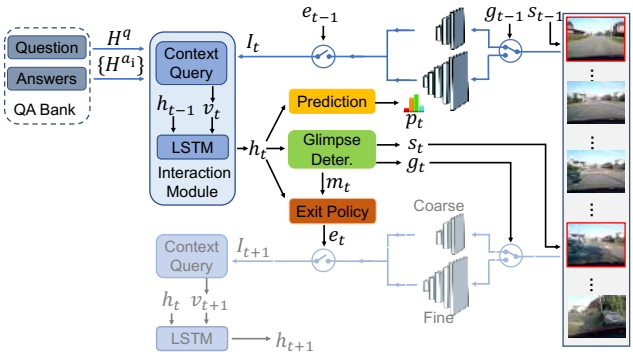


Figure 3. Architecture of Eclipse for dynamic causal reasoning.

Interaction Module. To assist the subsequent dynamic reasoning with rich guidance information, we design an Interaction Module to fuse different kinds of available inputs. This module, consisting of a context-query sub-module and an interaction LSTM, recurrently interacts with a small number of frames selected from the full video sequence containing T frames. As shown in Figure 3, at each inference step t, this module fuses the QA information (H^q and $\{H^{a_i}\}$), the currently selected frame (I_t), and the historical cues (h_{t-1}) to produce an expressive representation (h_t) that can be used for dynamic reasoning.

More formally, at the t^{th} inference step, to first fuse the textual QA information (H^q and $\{H^{a_i}\}$) with the currently selected visual frame feature (I_t), we use a context-query sub-module to perform such a fusion process. We implement this sub-module by following the context-matching module of [29], which can effectively fuse the visual feature sequence with the textual sequence to produce a combined representation. Since the original method in [29] takes a sequence of visual features as an input, and we have only the selected frame feature I_t here, we treat I_t as a visual feature sequence containing a single element. Hence the fusion process in this sub-module can be formulated as:

$$v_t^i = [I_t; F^{I_t,q}; F^{I_t,a_i}; I_t \odot F^{I_t,q}; I_t \odot F^{I_t,a_i}] \quad (1)$$

where $\odot$ represents element-wise product. $F^{I_t,q}$ and F^{I_t,a_i} are obtained by computing the similarity between the visual frame feature (I_t) and the textual features (H^q and H^{a_i}). More details of this process are referred to [29] and also our supplementary. For simplicity, we use v_t to represent the concatenated $\{v_t^i\}_{i=1}^N$, as shown in Figure 3. The output v_t, that incorporates the information of currently selected frame and the QA embedding, can be fed into the following interaction LSTM for more information interaction, as introduced below.

Besides fusing the QA information and the currently selected frame (I_t), to guide the dynamic reasoning at current inference step, we also incorporate the historical cues from past inference steps. Thus we design an interaction LSTM that takes the v_t as the new input to interact with the historical cues h_{t-1} as:

$$c_t, h_t = LSTM(v_t, c_{t-1}, h_{t-1}; \theta_{LSTM}) \quad (2)$$

where θ_{LSTM} are parameters of LSTM. The generated hidden state h_t encodes rich information of all available inputs and historical cues, and thus it can serve as an expressive representation to be fed into the downstream modules for dynamic reasoning as follows.

Prediction Module. This module is used to generate the reasoning result (i.e. the probability distribution over candidate answers) at the current inference step t. This module computes the reasoning result as: $p_t = f_p(h_t; \theta_p)$, where $p_t \in \mathbb{R}^N$ represents the probability scores for all candidate answers. f_p can be implemented with one fully-connected (FC) layer followed by a $Softmax$ classifier.

Glimpse-Determination Module. At each inference step t, conditioned on h_t, this module performs dynamic reasoning by making two decisions simultaneously. The first decision is to select which frame to be observed at next step, and the second is to decide whether to compute fine-grained features or coarse features for this selected frame. Corresponding to these two decisions, we design the following two branches within this module.

The skip-policy branch selects the frame that we need to skip to at next inference step via the following process: $s_t = f_s(h_t; \theta_s)$, where the output s_t indicates the decision of the next frame location. Note that our network can skip forth and back over the entire video sequence, which is conceptually similar to the human reasoning process where we need to not only jump forward to find future informative frames but also go back to examine past information.

Besides determining next frame, this module also has a granularity-policy branch that adaptively decides the feature computation granularity for the next selected frame, formulated as: $g_t = f_g(h_t; \theta_g)$. The output g_t, denotes the decision of feature granularity. In our implementation, we provide two kinds of feature granularity, namely, coarse features computed by a lightweight CNN; fine-grained features computed by a more representative yet computation-heavier CNN, to be chosen from. In the Glimpse-Determination module, both f_s and f_g are implemented with a FC layer.

Exit-Policy Module. To estimate when we can exit the reasoning process to achieve adaptive inference, we design the Exit-Policy Module. At each inference step t, this module decides if we can exit the reasoning process at current step based on the guidance information (h_t) as: $e_t = f_e(h_t; \theta_e)$, where the output e_t denotes the confidence score of terminating the reasoning at current step. By training the exit-policy, our network can achieve adaptive inference, such that only a small and flexible number of frames are selected and computed on a per-video basis to derive reliable reasoning result.

Optimization. To optimize the above modules in our

network, we introduce several loss functions. Specifically, at each inference step t, a cross-entropy loss $\mathcal{L}^t_{pred}$ is used to train the classifier of the Prediction Module:

$$\mathcal{L}^t_{pred} = -\sum_{n=1}^{N} y^n \log(p^n_t) \tag{3}$$

where y is the ground-truth one-hot label vector for the candidate answers and p^n_t is the predicted score for the n^{th} answer. As for Glimpse-Determination Module, to push the skip-policy branch to select a useful frame at each inference step, a simple yet effective loss is used:

$$\mathcal{L}^t_{incre} = -(m_t - m_{t-1}) \tag{4}$$

where $m_t = p^{gt}_t - \max\{p^{c'}_t \mid c' \neq gt\}$ is the margin between the predicted probability of the correct answer (indexed by gt) and the largest probability of other candidate answers. We can simply infer that a larger m_t indicates a more confident and accurate reasoning. Therefore, we use $m_t - m_{t-1}$ to encourage the margin to keep growing over the inference steps, which indicates that at each step, we aim to select a useful frame to benefit our dynamic reasoning, considering that the confidence of our network for the correct answer increases when seeing the selected frames.

Meanwhile, to further save computation cost and prevent the granularity-policy branch from constantly using the computation-heavy fine features, we penalize the feature granularity policy when the fine feature is computed at each inference step t as follows:

$$\mathcal{L}^t_{feat} = g_t \tag{5}$$

where $g_t = 1$ represents that the policy chooses to extract computation-heavy fine features for next step, while $g_t = 0$ means it switches to extract computation-cheap coarse features for next step. To optimize our whole Glimpse-Determination Module, we incorporate the above two loss functions, $\mathcal{L}^t_{incre}$ and $\mathcal{L}^t_{feat}$, into a combined loss:

$$\mathcal{L}^t_{glimpse} = \mathcal{L}^t_{incre} + \mathcal{L}^t_{feat} \tag{6}$$

Last but not least, we need to train the Exit-Policy to make a reliable decision if we can exit the reasoning process at current inference step. However, there are no ground-truth labels providing feedback on when our network can exit reasoning. Therefore, we leverage m_t to generate dynamic labels to train Exit-Policy. Recalling that m_t is the probability margin between the correct answer and the largest one of other candidate answers at each step, and m_t is optimized to keep increasing under the constraint of Eqn. 4. Therefore, the gap between m_t at different inference steps can be used to estimate the information gain of seeing more frames in our dynamic reasoning process. Given the pre-defined largest reasoning step T, the gap between m_t (at the current step) and m_T (of the final step) can estimate the value of remaining information gain by continuing reasoning till the end. Thus the gap can be used to determine whether our network can stop inference at t^{th} step. When m_t is very close to m_T, this means the information gain by observing more frames is small, and thus we can exit reasoning in advance to reduce computation without incurring decrease in prediction accuracy.

In particular, at each step t, if $m_T - m_t < \mu(m_T - m_1)$, which means m_t is close to m_T, i.e., the estimated remaining information gain is small enough, we set the label, y^t_{exit}, as 1, indicating our model can exit reasoning at the t^{th} inference step. Otherwise, the label is set to 0, representing we need to seek more frames. Here $\mu > 0$ controls how close m_t should be to m_T when the network exits inference. Conditioned on the estimated labels, y^t_{exit}, training Exit-Policy can be seen as a binary classification problem. Thus we train this module by minimizing a binary cross-entropy loss:

$$\mathcal{L}^t_{exit} = -[y^t_{exit} \log(e_t) + (1 - y^t_{exit}) \log(1 - e_t)] \tag{7}$$

By combining Eqns (3), (6), and (7), the total loss function for each step t can be formulated as:

$$\mathcal{L}^t = \mathcal{L}^t_{pred} + \mathcal{L}^t_{exit} + \lambda * \mathcal{L}^t_{glimpse} \tag{8}$$

where λ is the weight of the combined loss function for optimizing the Glimpse-Determination Module. In our experiments, we compute the sum: $\sum_{t=1}^{T} \mathcal{L}^t$ from all inference steps as the final optimization objective.

Note that Eqn. (8) cannot be optimized directly with gradient descent, since the involved decisions of selecting frames and determining feature granularity in our dynamic reasoning are discrete, and sampling from discrete distribution makes the network non-differentiable. To address this issue, we introduce an effective joint Gumbel-Softmax operation.

Joint Gumbel Softmax. The original Gumbel-Softmax Sampling [9] is an effective way to transform the original non-differentiable sample from a discrete distribution, to a differentiable decision from a corresponding Gumbel-Softmax distribution. In our task, to sample from the aforementioned two discrete distributions (namely, selecting frames and determining granularity) simultaneously, we here design an effective joint Gumbel-Softmax operation.

In particular, in the Glimpse-Determination Module, at each step t, we first derive the logits $z \in \mathbb{R}^{T*2}$ by feeding the hidden state h_t into a fully-connected layer. Then we use $Softmax$ to obtain a categorical distribution π_t from z: $\pi_t = \left\{p_{i,j} \mid p_{i,j} = \frac{\exp(z_{i,j})}{\sum_{c=1}^{T}\sum_{k=1}^{2}\exp(z_{c,k})}\right\}$. With the Gumbel-Max trick [9], the discrete sample from the categorical distribution π_t can be defined as follows:

$$\hat{l}_t = \underset{i\in\{1,\dots,T\}, j\in\{1,2\}}{\arg\max} (\log p_{i,j} + g_{i,j}) \tag{9}$$

where $g_{i,j} = -\log(-\log(u_{i,j}))$ denotes the Gumbel noise, and $u_{i,j}$ is the i.i.d. samples drawn from $Uniform(0,1)$. We can further relax the non-differentiable operation $argmax$ with $softmax$ to facilitate gradient-based optimization:

$$l_t = \left\{ P_{i,j} \mid P_{i,j} = \frac{\exp\left((\log p_{i,j} + g_{i,j})/\tau\right)}{\sum_{c=1}^{T}\sum_{k=1}^{2}\exp\left((\log p_{c,k} + g_{c,k})/\tau\right)} \right\}, \quad for\ i \in \{1,...,T\}, j \in \{1,2\} \tag{10}$$

where τ is the temperature parameter, which controls the smoothness of the sampling mechanism. When $\tau \to 0$, the sampling approximates the $argmax$ operation in Eqn. 9. The output l_t incorporates the output of two decisions: the first dimension of l_t denotes the decision of selecting the frame (i.e, the output s_t of the skip-policy branch) and the second dimension of l_t denotes the decision of the feature granularity of the selected frame (i.e, the output g_t of the granularity-policy branch).

By using the outputs (s_t and g_t) of the joint Gumbel-Softmax operation, our network manages to dynamically select the frame for next inference step and specify the feature granularity for the selected frame at each step. Therefore, by introducing the joint Gumbel-Softmax, our network can learn the two discrete policy decisions jointly in a fully differentiable way.

Training and testing. During training, we optimize our network within a fixed number of steps, which means the exit-policy is trained together with other modules but the exit decisions are not used. However, at the testing phase, if the exit-policy decides to stop reasoning at the t^{th} inference step, we exit the model and use the current prediction result, p_t, as the final reasoning result. In such a manner, our model achieves dynamic causal reasoning.

5. Experiments

Given that the number of candidate answers for each question is not fixed in our dataset, we evaluate the performance of our network using binary and multi-choice setups. In binary case (denoted as **Setting-1/2**), the input to the model is a question with an answer, and the model needs to predict the correctness of this answer. In multi-choice setup (denoted as **Setting-1/4**), models are expected to select the correct answer from 4 candidate answers (i.e, 3 of them are incorrect). These two experiment setups can be treated as binary and four-class classification problems.

Implementation Details. We compute features from the penultimate layer of a pretrained ResNet-101 model [19] as the fine-grained frame feature, and a pretrained MobileNetv2 [41] is used as the lightweight CNN to extract coarse features. In the QA Bank, we use Glove [37] to embed QA text, and then use a BiLSTM with 150-Dimension hidden states to encode the textual sequence. As for the Interaction LSTM, the dimension of hidden states is 300. We implement the framework using Pytorch and adopts Adam [26] with a learning rate of 3e-4 and a weight-decay of 1e-5. The μ in the Exit-Policy is set to 0.1 and λ is set to 0.01 in the loss function. We follow [9] and set the initial temperature τ to 5, and gradually anneal it with an exponential decay factor of -0.045 in every epoch. According to evaluation statistics, our network shows a very fast **inference speed** of $16ms$ per testing video on a Nvidia RTX 2080Ti GPU.

We compare our network with the following baselines. **Text-only models.** These models only relying on text information without visual input, are relatively weak baselines used to assess language biases in our SUTD-TrafficQA. **Q-type (random)** randomly selects an answer from the answer space. **QE-LSTM** uses Glove [37] to embed the input question and then encode it with LSTM [20]. The final LSTM hidden state is passed to a MLP for predicting the correct answer. Different from **QE-LSTM** using questions only, **QA-LSTM** uses LSTM to encode both question embedding and answer embedding, and the final hidden states are used for predicting the answer.

Text+video models. We evaluate the following models that require both video and text inputs. **VIS+LSTM** [38] uses LSTM to encode image representation and textual features. Since the original method takes a single image as input, we adapt this method by averaging features of all sampled frames in a video as the visual input. **Avgpooling** uses each frame with the encoded QA features to compute a prediction for each frame. We then perform mean pooling over all the frame predictions to obtain the final result. **CNN+LSTM** uses two LSTMs to encode both the video sequence and the QA text respectively. The two final hidden states are concatenated to predict the correct answer. **I3D+LSTM** uses I3D network [4] to extract video motion features, and then fuse with QA textual features encoded by LSTM to compute the model prediction. **TVQA** [29] is a multi-stream network to fuse input features from different modalities to answer the question. **HCRN** [10] adopts a Hierarchical Conditional Relation Networks to model sophisticated structure for reasoning over videos. **BERT-VQA** [55] uses BERT [6] to encode the visual and language information jointly to predict the answer.

5.1. Results and Analysis

The results in Table 2 demonstrate the minimal language biases in SUTD-TrafficQA, as the text-only baselines perform almost the same as the random choice. In contrast, the models using video input achieve obviously higher accuracy than text-only baselines. This demonstrates that to solve the reasoning tasks in our dataset, the model needs to associate visual content with linguistic cues to infer correct

Figure 4. A qualitative example. The numbers above the selected frames show the order of the sequence selected by our network. It shows that our model selects informative frames dynamically and allocates large computation budget using fine features to most relevant frames for causal reasoning. More examples in supplementary.

Table 2. Results on SUTD-TrafficQA dataset.

Models	Setting-1/4	Setting-1/2	GFLOPs
Q-type (random)	25.00	50.00	-
QE-LSTM	25.21	50.45	-
QA-LSTM	26.65	51.02	-
Avgpooling	30.45	57.50	252.69
CNN+LSTM	30.78	57.64	252.95
I3D+LSTM	33.21	54.67	108.72
VIS+LSTM [38]	29.91	54.25	252.80
BERT-VQA [55]	33.68	63.50	266.77
TVQA [29]	35.16	63.15	252.11
HCRN [10]	36.49	63.79	2051.04
Eclipse	**37.05**	**64.77**	**28.14**
Human	95.43	96.78	-

Table 3. Results of removing granularity-policy in Eclipse.

Models	Coarse Features Only	Fine Features Only	Eclipse (dynamic granularity)
Accuracy	34.35	37.16	37.05
GFLOPs	10.61	133.75	28.14

Table 4. Results of removing skip-policy in Eclipse. Uniform-n means uniformly sampling n frames from the video for reasoning.

Models	Uniform-10	Uniform-20	Uniform-40	Eclipse (skip-policy)
Accuracy	34.16	35.49	36.48	37.05
GFLOPs	32.17	51.19	68.41	28.14

Table 5. Results of removing exit-policy in Eclipse. Final-Step Inference refers to that we remove the exit-policy and infer until the final step.

Models	Final-Step Inference	Eclipse (exit-policy)
Accuracy	37.11	37.05
GFLOPs	36.92	28.14

answers. For a fair computation-efficiency comparison, we pick models requiring video input to compute GFLOPs per video, as the visual feature extraction consumes much computation budget, and the metric of GFLOPs is independent of hardware configurations. The results show our Eclipse achieves state-of-the-art reasoning accuracy with significantly improved computation efficiency. This verifies that compared to conventional video QA methods, through dynamic causal reasoning, our model effectively exploits the spatio-temporal and logical structure of video events to infer correct answers with much smaller frame usage and efficient feature computation. In addition, three volunteers who did not see the videos and questions before, were invited to pick correct answers, and we use the average prediction accuracy as *Human* performance. The discrepancy between neural networks and the human performance demonstrates the challenging nature of our dataset and the necessity of further research in video reasoning area.

Ablation Study. As shown in Table 3, compared with the model variant of using fine features only, by adopting the granularity-policy to adaptively decide feature granularity for the selected frame at each step, our network achieves nearly the same accuracy yet using much lower computation cost. Our network also achieves obviously higher reasoning accuracy than the method of using coarse features only. These results show the effectiveness of our granularity-policy by choosing fine features for most useful frames and coarse features for less important frames. Furthermore, we remove the skip-policy and simply uniformly sample frames at each step. As shown in Table 4, our Eclipse performs the best yet at small computation cost. This shows that our skip-policy effectively reduces the computation cost by selecting useful frames for dynamic reasoning. Moreover, we present the frame location distributions for the first three inference steps of our network in Figure 5. As shown, our network selects frames dynamically for different videos as we expect. We also investigate the exit-policy module by comparing it with the method of reasoning until the final inference step. The result in Table 5 shows that our model achieves best accuracy-to-computation ratio via adaptive inference. In Figure 4, we further present a qualitative example from our dataset to show how Eclipse performs dynamic and efficient reasoning.

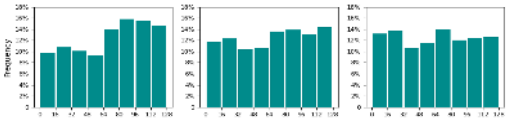


Figure 5. Distributions of frame location (Left to right:step1, 2, 3).

6. Conclusion

We create a new video QA dataset, SUTD-TrafficQA, focusing on video reasoning over traffic events. In our dataset, we introduce 6 reasoning tasks requiring various levels of causal reasoning. Besides, we propose the Eclipse network for video QA. By learning the dynamic glimpse policy and adaptive exit policy, our network achieves superior performance with significant computation efficiency.

Acknowledgement. We would like to thank Yutian Lin, Renhang Liu, Yingjie Qiao, Xun Long Ng, Tran Nguyen Bao Long, Koh Kai Ting and Christabel Dorothy for their help in dataset collection and running baseline models. This work is supported by SUTD Projects PIE-SGP-AI2020-02 and SRG-ISTD-2020-153.

References

[1] Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc V Gool. Soft-to-hard vector quantization for end-to-end learning compressible representations. In *Advances in Neural Information Processing Systems*, pages 1141–1151, 2017. 3

[2] Michele Banko and Eric Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, page 26–33, USA, 2001. Association for Computational Linguistics. 1

[3] Shweta Bhardwaj, Mukundhan Srinivasan, and Mitesh M. Khapra. Efficient video classification using fewer frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 7

[5] Rohan Chandra, Tianrui Guan, Srujan Panuganti, Trisha Mittal, Uttaran Bhattacharya, Aniket Bera, and Dinesh Manocha. Forecasting trajectory and behavior of road-agents using spectral clustering in graph-lstms. *IEEE Robotics and Automation Letters*, 5(3):4882–4890, 2020. 2

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 7

[7] Geoffrey B Duggan and Stephen J Payne. Skim reading by satisficing: evidence from eye tracking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1141–1150, 2011. 4

[8] C. Fan et al. Heterogeneous memory enhanced multimodal attention model for video question answering. In *CVPR*, 2019. 2

[9] J. Eric et al. Categorical reparameterization with gumbel-softmax. *arXiv*, 2016. 6, 7

[10] Le. Minh et al. Hierarchical conditional relation networks for video question answering. In *CVPR*, 2020. 1, 4, 7, 8

[11] X. Li et al. Beyond rnns: Positional self-attention with co-attention for video question answering. In *AAAI*, 2019. 2

[12] Y. Jang et al. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *CVPR*, 2017. 2, 3, 4

[13] Z. Amir et al. Social-iq: A question answering benchmark for artificial social intelligence. In *CVPR*, 2019. 2, 3

[14] Michael Figurnov, Maxwell D Collins, Yukun Zhu, Li Zhang, Jonathan Huang, Dmitry Vetrov, and Ruslan Salakhutdinov. Spatially adaptive computation time for residual networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1039–1048, 2017. 2

[15] Gemma Fitzsimmons, Mark J Weal, and Denis Drieghe. Skim reading: an adaptive strategy for reading on the web. In *Proceedings of the ACM Conference on Web Science*, pages 211–219, 2014. 4

[16] Noa Garcia and Yuta Nakashima. Knowledge-based video question answering with unsupervised scene descriptions. *arXiv preprint arXiv:2007.08751*, 2020. 2

[17] Noa Garcia, Mayu Otani, Chenhui Chu, and Yuta Nakashima. Knowit vqa: Answering knowledge-based questions about videos. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020. 2

[18] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24:8–12, 2009. 1

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7

[20] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 7

[21] Drew Arad Hudson and Christopher D. Manning. Compositional attention networks for machine reasoning. In *International Conference on Learning Representations*, 2018. 4

[22] Huaizu Jiang, Ishan Misra, Marcus Rohrbach, Erik Learned-Miller, and Xinlei Chen. In defense of grid features for visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[23] Junyeong Kim, Minuk Ma, Trung Pham, Kyungsu Kim, and Chang D. Yoo. Modality shifting attention network for multi-modal video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2

[24] Kyung-Min Kim, Seong-Ho Choi, Jin-Hwa Kim, and Byoung-Tak Zhang. Multimodal dual attention memory for video story question answering. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2

[25] Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. Deepstory: Video story qa by deep embedded memory networks. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2016–2022, 2017. 2

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[27] Bruno Korbar, Du Tran, and Lorenzo Torresani. Scsampler: Sampling salient clips from video for efficient action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6232–6242, 2019. 2

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 2

[29] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. Tvqa: Localized, compositional video question answering. *arXiv preprint arXiv:1809.01696*, 2018. 2, 3, 4, 5, 7, 8

[30] Jie Lei, Licheng Yu, Tamara Berg, and Mohit Bansal. TVQA+: Spatio-temporal grounding for video question answering. In *Proceedings of the 58th Annual Meeting of*

the Association for Computational Linguistics, pages 8211–8225, Online, July 2020. Association for Computational Linguistics. 1, 2

[31] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *International Conference on Learning Representations*, 2016. 3

[32] Junwei Liang, Lu Jiang, Liangliang Cao, Li-Jia Li, and Alexander G Hauptmann. Focal visual-text attention for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6135–6143, 2018. 2

[33] Yihang Lou, Yan Bai, Jun Liu, Shiqi Wang, and Lingyu Duan. Veri-wild: A large dataset and a new method for vehicle re-identification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3235–3243, 2019. 2

[34] Tegan Maharaj, Nicolas Ballas, Anna Rohrbach, Aaron Courville, and Christopher Pal. A dataset and exploration of models for understanding video data through fill-in-the-blank question-answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6884–6893, 2017. 2

[35] Guodong Mu, Di Huang, Guosheng Hu, Jia Sun, and Yunhong Wang. Led3d: A lightweight and efficient deep approach to recognizing low-quality 3d faces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[36] Jonghwan Mun, Paul Hongsuck Seo, Ilchae Jung, and Bohyung Han. Marioqa: Answering questions by watching gameplay videos. In *ICCV*, 2017. 3

[37] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *In EMNLP*, 2014. 7

[38] Mengye Ren, Ryan Kiros, and Richard Zemel. Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961, 2015. 7, 8

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2

[40] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. Movie description. *Int. J. Comput. Vision*, 123(1):94–120, May 2017. 2

[41] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 7

[42] Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. Green ai. *arXiv preprint arXiv:1907.10597*, 2019. 2, 4

[43] Xiaomeng Song, Yucheng Shi, Xin Chen, and Yahong Han. Explore multi-step reasoning in video question answering. In *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, 2018. 2

[44] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in nlp. *arXiv preprint arXiv:1906.02243*, 2019. 2, 4

[45] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015. 2

[46] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Sparsifying neural network connections for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4856–4864, 2016. 3

[47] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4631–4640, 2016. 2, 3

[48] Yao-Hung Hubert Tsai, Santosh Divvala, Louis-Philippe Morency, Ruslan Salakhutdinov, and Ali Farhadi. Video relationship reasoning using gated spatio-temporal energy graph. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 2

[50] Bo Wang, Youjiang Xu, Yahong Han, and Richang Hong. Movie question answering: Remembering the textual cues for layered visual contents. *arXiv preprint arXiv:1804.09412*, 2018. 2

[51] Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E Gonzalez. Skipnet: Learning dynamic routing in convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 409–424, 2018. 3

[52] Zuxuan Wu, Caiming Xiong, Yu-Gang Jiang, and Larry S Davis. Liteeval: A coarse-to-fine framework for resource efficient video recognition. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 2

[53] Zuxuan Wu, Caiming Xiong, Chih-Yao Ma, Richard Socher, and Larry S Davis. Adaframe: Adaptive frame selection for fast video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1278–1287, 2019. 2

[54] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *ACM Multimedia*, 2017. 2, 3

[55] Zekun Yang, Noa Garcia, Chenhui Chu, Mayu Otani, Yuta Nakashima, and Haruo Takemura. Bert representations for video question answering. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020. 7, 8

[56] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, 2019. 2, 3

[57] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In

Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4651–4659, 2016. 2

[58] Tackgeun You and Bohyung Han. Traffic accident benchmark for causality recognition. In *Proceedings of the European Conference on Computer Vision*, 2020. 2

[59] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2

[60] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. End-to-end concept word detection for video captioning, retrieval, and question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3165–3173, 2017. 2

[61] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *AAAI*, pages 9127–9134, 2019. 2

[62] Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Tensor fusion network for multimodal sentiment analysis. In *Empirical Methods in Natural Language Processing, EMNLP*, 2017. 2

[63] Kuo-Hao Zeng, Tseng-Hung Chen, Ching-Yao Chuang, Yuan-Hong Liao, Juan Carlos Niebles, and Min Sun. Leveraging video descriptions to learn video question answering, 2016. 2

[64] J. Zhang, F. Wang, K. Wang, W. Lin, X. Xu, and C. Chen. Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1624–1639, 2011. 1

[65] Z. Zhao, Z. Zhang, X. Jiang, and D. Cai. Multi-turn video question answering via hierarchical attention context reinforced networks. *IEEE Transactions on Image Processing*, 28(8):3860–3872, 2019. 2

[66] Linchao Zhu, Zhongwen Xu, Yi Yang, and Alexander G Hauptmann. Uncovering the temporal context for video question answering. *International Journal of Computer Vision*, 124(3):409–421, 2017. 1, 2