Knowledge Proxy Intervention for Deconfounded Video Question Answering

Jiangtong Li1, Li Niu1*, Liqing Zhang1*

1 Department of Computer Science and Engineering, MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University

{keep_moving-Lee,ustcnewly,lqzhang}@sjtu.edu.cn

Abstract

Recently, Video Question-Answering (VideoQA) has drawn more and more attention from both the industry and the research community. Despite all the success achieved by recent works, dataset bias always harmfully misleads current methods focusing on spurious correlations in training data. To analyze the effects of dataset bias, we frame the VideoQA pipeline into a causal graph, which shows the causalities among video, question, aligned feature between video and question, answer, and underlying confounder. Through the causal graph, we prove that the confounder and the backdoor path lead to spurious causality. To tackle the challenge that the confounder in VideoQA is unobserved and non-enumerable in general, we propose a model-agnostic framework called Knowledge Proxy Intervention (KPI), which introduces an extra knowledge proxy variable in the causal graph to cut the backdoor path and remove the effect of confounder. Our KPI framework exploits the front-door adjustment, which requires no prior knowledge about the confounder. The effectiveness of our KPI framework is corroborated by three baseline methods on five benchmark datasets, including MSVD-QA, MSRVTT-QA, TGIF-QA, NExT-QA, and Causal-VidQA.

1. Introduction

In recent years, Video Question-Answering (VideoQA) has drawn more attention from the industry and research community due to its essential role in interactive artificial intelligence and recognition science. In VideoQA, there are three crucial challenges, (1) how to capture the visual clues in the video (e.g., object, action, and causality), (2) how to parse the semantics and syntax in language, and (3) how to align the visual clue with the linguistic semantics and syntax. Therefore, lots of works [12, 25, 22, 62, 35, 34, 5] have studied the VideoQA from these three aspects, and have also achieved great success in both open-ended VideoQA [66, 24] and multi-choice VideoQA [24, 61, 32].

As the core of VideoQA, video (V), question (Q), and the aligned feature between video and question (aligned feature for short, H) play essential roles in predicting answer (A). However, due to dataset bias, most of existing methods, which target at predicting answers directly from the observational probability \( P(A|V, Q, H) \), will be inevitably misled to spurious correlation, and have trouble in revealing the causal relation between the V, Q, H, and A. In Figure 7, we show two examples to explain how dataset bias affects the answer prediction. For example, in Figure 7 (a), since the kangaroo can rarely appear indoors, the model would ignore the “unique jumping pose” and the “distinct wobble of tail” from the kangaroo, and regard it as a cat. Furthermore, dataset bias is from nature (Zipf’s law [60] and social conventions [19]), i.e., more cats are indoors, and more kangaroos are outdoors. Therefore, simply enlarging the dataset would never eliminate dataset bias. To this end, we focus on dataset bias in VideoQA task and exploit the concepts of confounder to analyze and alleviate this problem.

The causal graph of the VideoQA pipeline is illustrated in Figure 2 (a), where \( V, Q, H, A, \) and \( C \) represent video, question, aligned feature, answer, and confounder, respec-
In this paper, we propose KPI framework, an implementation of front-door adjustment, which is model-agnostic and can help current methods to mitigate spurious correlations from dataset bias. In particular, given that knowledge proxy and its representation are not pre-defined, we propose a series of practical approximations in Section 4. The effectiveness of KPI framework is corroborated by comprehensive experiments with three baseline methods (CoMem [12], HGA [26], and HQGA [62]) on five benchmark datasets (MSVD-QA [65], MVRVT-TQA [65], TGIQA [24], NExTQA [61], and Causal-VidQA [32]). Our main contributions are summarized as follows:
• We focus on dataset bias and provide a thorough analysis of how dataset bias affects the answer prediction using the causal graph.

• To alleviate the effect of dataset bias, we exploit front-door adjustment and propose our model-agnostic KPI framework to implement the causal intervention.

• Comprehensive experiments with three baseline methods on five benchmark datasets reveal that our framework significantly boosts the state-of-the-art methods.

2. Related Work

2.1. Video Question Answering

VideoQA, as the core of visual-language representation [31, 14, 30, 3, 37, 4] and reasoning [39, 69, 35, 34], aims to answer the question based on dynamic visual content. To this end, the VideoQA benchmarks start from the problem of description [66, 24, 29] and then build more challenging datasets towards temporal reasoning [6], physical reasoning [71], evidence reasoning [61], and commonsense reasoning [32]. Although the architecture of VideoQA methods has changed significantly in recent years, the core of VideoQA methods is still video representation, question representation, and video-question aligned representation. For video representation, early efforts [24, 12] usually exploit the appearance feature [18] and motion feature [64] along with Recurrent Neural Network (RNN) [20] or Transformer [54]. As the development of object-level representation, MIN [27] and MASN [51] introduce the bounding-box feature into video representation. For question representation, most existing works utilize word embedding [47] along with RNN. As the improvement of pre-trained language model, BERT feature [8] is exploited by NExT-QA [61] and then becomes widely used in recent works [62, 35, 34, 63]. For video-question aligned representation, early efforts tend to implement alignment through cross-modal attention [33, 13] or memory network [12, 9]. As the graph models are introduced into VideoQA, graph reasoning [22, 26, 40, 38, 15, 55, 5] is explored more in video-question alignment. Recently, the natural hierarchical structure [28, 16, 45, 7, 46] of video, i.e., object-appearance-motion and appearance-motion, also draws more and more attention. Among them, HCRN [28] proposes conditional relation block and stacks it to capture information from different video intervals, whereas MSPAN [16] establishes cross-scale feature interaction on top of the hierarchy. HQGA [62] and VGT [63] align question and video hierarchy from low-level visual entities to high-level activities.

Some works also look into the scene bias [35, 34] or atemporal VideoQA [2], which focuses on the observed and enumerable bias. Unlike them, we are the first to study dataset bias in VideoQA from a general viewpoint and require no prior knowledge about the confounder.

2.2. Causal Inference

Causal inference [43, 50] provides us with a powerful tool to analyze the dataset bias and mitigate spurious correlations, which can be divided into deconfounding [57, 73, 70, 69, 36] and counterfactual inference [10, 74, 58, 39]. Besides, it has been used in various learning tasks, including image classification [57], image segmentation [73], image caption [70, 36], image question answering [39], language understanding [10], dialogue system [74], and recommendation system [58], which not only enables deep learning methods with the ability to learn causal effects but also boosts the performance of current methods. The generic way is to disentangle all variables in the target task and model the causal effects among variables on causal graph.

Some works also study front-door adjustment [70, 69], which focus on either description towards image [69] or confounding effect within models, like Transformer [70]. Different from them, we focus on description, evidence reasoning, and commonsense reasoning towards video and are the first to apply front-door adjustment to mitigate the spurious correlations within dataset bias in VideoQA.

3. Causal Intervention

In this section, we introduce the concepts of causal inference [41, 44], including the confounder (Section 3.1), the backdoor adjustment (Section 3.2), and the front-door adjustment (Section 3.3). In the following sections, we use boldface lower letter, (v, q, h, z), to represent the feature vector, boldface capital letter, (V, Q, H, Z), to represent the feature space, and the calligraphic letter, (\(\mathcal{V}, \mathcal{Q}, \mathcal{H}, \mathcal{Z}\)), to represent the variable in the causal graph. More background and detailed derivation are in Supplementary Material.

3.1. Confounder

The observational probability can be formulated as

\[
P(A|V, Q, H) = \sum_c P(A|V, Q, H, c)P(c|V, Q, H),
\]

where c is the split of the confounder, like the environment or the action. During training, it is much easier for current methods to recognize some of the video and question concepts, and ignore the characteristic of other video and question concepts. Therefore, during inference, current methods tend to directly predict the answer based on co-occurrences with those recognized concepts instead of reasoning from the videos and questions, i.e., a partition in \(\mathcal{C}\) dominates the \(P(A|V, Q, H)\) by \(P(c|V, Q, H)\). For example, in Figure 7 (b), since run co-occurs much more with jump than shoot, once the model detects run, it would predict the answer as jump without noticing the fallen board or the dirt.
adjustment can also be used to implement 3.3. Front-door Adjustment

The technique of **do-calculus** is introduced in [43, 44]. Specifically, \(\text{do}(V, Q, H)\) denotes that we actively assign values to variable \(V, Q, H\), rather than passively observe them. As illustrated in Figure 2 (b), \(\text{do}(V, Q, H)\) indicates that we need to cut all incoming arrows to \(V, Q, H\), and make the \(V, Q, H\) independent to the confounder \(C\). Note that, all backdoor path from \(V, Q, H\) to \(A\) is from \(C \rightarrow Q\), and **do-calculus** only needs to cut \(C \rightarrow Q\) to prevent the backdoor path \(Q \leftarrow C \rightarrow A\) and \(A \leftarrow Q \leftarrow C \rightarrow A\). Therefore, the formulation of \(P(A|\text{do}(V, Q, H))\) is derived as

\[
P(A|\text{do}(V, Q, H)) = \sum_c P(A|\text{do}(V, Q, H), c)P(c|\text{do}(V, Q, H));
\]

\[
= \sum_c P(A|V, Q, H, c)P(c).
\]

The formulation of backdoor adjustment is intuitive and elegant. However, this formulation requires observing and enumerating all the factors in the confounder. Since dataset bias is very complex, it is impossible to disentangle all factors within the confounder. For example, in Figure 7, we can find two kinds of biases, i.e., the environment bias and the action bias, each of which contains many concrete items. Besides, dataset bias is not brought by only one type of bias independently but more likely by the combinations of different types of biases, like run indoors, run outdoors, jump indoors, jump outdoors, etc. Furthermore, using pre-trained model without pre-training data also prevents us from realizing potential confounder. Therefore, it is nearly impossible to get a reasonable split of the confounder \(C\) for backdoor adjustment.

3.3. **Front-door Adjustment**

Different from the backdoor adjustment, front-door adjustment can also be used to implement \(P(A|\text{do}(V, Q, H))\), with which we do not need to split the confounder \(C\). As illustrated in Figure 2 (c), to apply the front-door adjustment, an additional intermediate variable \(Z\) should be inserted between \(Q, H\) and \(A\) to construct front-door paths \(Q \rightarrow Z \rightarrow A\) and \(H \rightarrow Z \rightarrow A\). The causal intervention is then decomposed into two parts:

\[
P(A|\text{do}(V, Q, H)) = \sum_Z P(Z|\text{do}(V, Q, H))P(A|\text{do}(Z)).
\]

The first term in front-door adjustment is formulated as

\[
P(Z|\text{do}(V, Q, H)) = P(Z|V, Q, H) = P(Z|Q, H),
\]

and the second term is formulated as

\[
P(A|\text{do}(Z)) = \sum_v \sum_q \sum_h P(A|z, q, h, v)P(q, h, v),
\]

\[
= \sum_v \sum_q \sum_h P(A|z, q, h, v)P(v)P(q|v)P(h|q, v),
\]

where \(v, q, h\) represents all the possible representations in video, question, and aligned feature space.

To sum up, by applying Equation 4 and 5 into Equation 3, we have the front-door adjustment:

\[
P(A|\text{do}(V, Q, H)) = \sum_v \sum_q \sum_h P(v)P(q|v)P(h|q, v)\sum_z P(z|Q, H)P^*(A),
\]

\[
= \mathbb{E}_v\mathbb{E}_{q|v}\mathbb{E}_{h|q,v}\mathbb{E}_{z|Q,H}[P^*(A)],
\]

where \(P^*(A) = P(A|v, q, h, z)\). By applying the Normalized Weights Geometric Mean (NWGM) [53, 67], the outer expectation is moved into feature level:

\[
P(A|\text{do}(V, Q, H)) = \mathbb{E}_v\mathbb{E}_{q|v}\mathbb{E}_{h|q,v}\mathbb{E}_{z|Q,H}[P(v|q, h, z)]
\]

\[
\approx \text{Softmax}[g(E_v, E_{q|v}, E_{h|q,v}, E_{z|Q,H}[z])],
\]

where \(g(\cdot)\) is a fully-connect layer.

So far, we have introduced the reason for dataset bias (i.e., the confounder \(C\) and the backdoor path) and the theoretical solution: front-door adjustment.

4. **Methodology**

In this section, we will introduce the implementation of Equation 7 from two aspects, the knowledge space along with the other three feature spaces (Section 4.1) and the approximation of the expectation (Section 4.2). In Section 4.3, we will introduce the overall pipeline of KPI framework.

4.1. **Knowledge Space and Feature Spaces**

**Knowledge Space Z.** As explained in Section 1, the VideoQA model would be misled by dataset bias and ignore the causal relation between video-question and the answer. Therefore, in knowledge space \(Z\), we aim to separate the causal relations from correlations. Towards this end, we propose to first extract the correlated concepts from video-questions and answers, and then select the causal relations with existing knowledge graphs. In detail, we propose to build the knowledge space \(Z\) in the following steps,

1. For each training instance, we extract the actions and objects from video with 1B3 ResNeXt-101 [64, 17] and
Faster R-CNN [49] as video concepts \((C_v)\), extract the key words and phrases with NLTK [1] from question as question concepts \((C_q)\). Besides, we extract key words and phrases with NLTK [1] for multi-choice answers, and directly keep the answer for open-ended answers as answer concepts \((C_a)\).

2. For each training instance, we generate the correlated concepts (head-tail, \(h\)-\(t\)) from \(C_v\), \(C_q\), and \(C_a\), where \(h\) \(\in C_v \cup C_q\), and \(t\) \(\in C_a\).

3. For all training instances, we collect all the correlated concepts to initialize knowledge space \(Z\).

4. For each correlated concept \((h\_t)\) in knowledge space, if the \(h\) and \(t\) are adjacent nodes in existing knowledge graphs, the correlated concept is expanded with the node relation as a causal concept (head-relation-tail, \(h\_r\_t\)); otherwise, it is removed from knowledge space.

5. For each causal concept in knowledge space, we transform it into trainable knowledge embedding vectors with pre-trained BERT [8].

More details about knowledge space are in Supplementary. For knowledge graphs, we explore ConceptNet [52] and Atomic [23] to select the causal concepts, where ConceptNet focuses on physical-entity relations and Atomic concentrates on event-centered and social-interaction relations.

Furthermore, each knowledge embedding vector from the knowledge space cannot solely emphasize the information from the video and the question or infer the answer. However, combining multiple knowledge embedding vectors, the knowledge space could provide enough clues to summarize \(Q\) and \(H\) and reflect the causal relations for question answering simultaneously. To this end, the KPI framework will first use the question features and aligned features to softly retrieve the related knowledge embedding vectors \((Q \rightarrow Z \leftarrow H)\) and then exploit these knowledge embedding vectors to predict the answer \((Z \rightarrow A)\).

**Video Feature Space \(V\).** For each video, there are three types of features exploited in current methods, i.e., the motion feature from clips, the appearance feature from frames, and the bounding-box feature from objects. For each type of feature, we first collect all the feature vectors from the whole training set based on different feature extractors (i.e., I3D ResNeXt-101 for motion feature, ResNet-152 for appearance feature, and Faster R-CNN for bounding-box feature), and then apply the k-means algorithm to reduce the number of feature vectors within each type of video sub-embeddings to \(k_v\). Therefore, the video feature space has three sub-spaces, i.e., the motion feature space \(V_m\), the appearance feature space \(V_a\), and the bounding-box feature space \(V_o\). Since different baseline methods use different types of video features, the video feature space is decided by the input of each method. For example, for CoMem [12] and HGA [26], \(V\) is the appearance feature space \(V_a\) and the motion feature space \(V_m\); for HQGA [62], \(V\) is the bounding-box feature space \(V_o\), the appearance feature space \(V_a\), and the motion feature space \(V_m\).

**Question Feature Space \(Q\).** For question feature space, we first extract question features with fine-tuned BERT. For each question, the question feature \(Q_i \in \mathbb{R}^{n_q \times d_q}\) is then average-pooled along the question sequence to get the question vector \(q_i \in \mathbb{R}^{d_q}\). Finally, the k-means algorithm is applied to reduce the number of feature vectors to \(k_Q\).

**Aligned Feature Space \(H\).** The aligned feature is generated from video-question interaction, which cannot be directly extracted from uni-modal pre-trained model. To build the aligned feature space, we first rely on the baseline method by training a baseline model on observation probability, and then inferring the aligned vector for each video-question pair with the trained model. Like the video feature space and question feature space, the k-means algorithm is also applied to reduce the number of vectors into \(k_H\).

### 4.2. Expectation

In Equation 7, we need to calculate \(E_v[v]\), \(E_{[q,v]}[q]\), \(E_{[h,q,v]}[h]\), and \(E_{[x,Q,H]}[z]\), each of which is an approximation to the expectation in corresponding feature space. Here we use \(E_{[x,Q,H]}[z]\) as an example to show how the expectation is calculated with the EXP module. Given the knowledge space \(Z\), we have \(E_{[x,Q,H]}[z] = \sum_z P(z|Q,H)z\),
where the conditional distribution \( P(z|Q, H) \) can be approximated by attention modules. Specifically, we explore three different kinds of attention mechanisms for this approximation, including channel attention \([21]\), product attention \([59]\), and multi-head attention \([54]\). The inputs of each attention module are the knowledge space \( Z = [z_1, ..., z_{kZ}] \), and the concatenation of the question embedding and the aligned embedding \( \text{Cat}(q, h) \) from \( Q \) and \( H \).

For channel attention, it can be formulated as

\[
a_i = w^T \tanh(W_1z_i + W_2\text{Cat}(q, h)),
\]

\[
\alpha = \text{softmax}(a), z = \sum_{i=1}^{kZ} \alpha_i z_i,
\]

where \( w^T, W_1 \) and \( W_2 \) are trainable parameters. For product attention, it can be formulated as

\[
z = \text{softmax}(\text{Cat}(q, h)W_1(ZW_2)^T/\sqrt{d_z})ZW_3,
\]

where \( W_1, W_2, \) and \( W_3 \) are trainable parameters. For multi-head attention, it is formulated as

\[
\text{head}^k = \text{softmax}(\text{Cat}(q, h)W_1^k(ZW_2^k)^T/\sqrt{d_z})ZW_3^k,
\]

\[
z = \text{Cat}(\text{head}^1, ..., \text{head}^8)W_{out},
\]

where \( W_1^k, W_2^k, W_3^k, W_{out} \) are trainable parameters, and we exploit eight heads here.

Note that although \( E_v[v] \) does not condition on any variable, we still need to approximate \( E_v[v] \) via \( E_{[v]v}[v] \) for each training instance. Otherwise, the approximated results will degrade into a single fixed vector for all different inputs. Expressly, for each video with different input features, \( i.e. \), motion, appearance, and bounding-box, we first adopt a self-attention layer along with average pooling to each type of video feature independently to get the video sub-embeddings, \( i.e. \), \( \hat{v}_m \), \( \hat{v}_a \), and \( \hat{v}_o \). Then we calculate \( E_{[v]v}[v] \) on each video sub-space with the corresponding video sub-embedding. More details about the video sub-embeddings can be found in Supplementary Material.

### 4.3. Knowledge Proxy Intervention

The structure of our Knowledge Proxy Intervention (KPI) framework is illustrated in Figure 4. Given the video and question, the baseline method (with the video and the question feature extractor) is utilized to get the video sub-embeddings, the question embedding, and the aligned embedding. Then the knowledge space, \( Z \), along with the question embedding and the aligned embedding, is sent into the EXP module to calculate \( E_{[z]Q,H}[z] \). Meanwhile, the video feature space \( V \) (with the video sub-embeddings), the question feature space \( Q \), and the aligned feature space \( H \) are sent into different EXP modules in turn to calculate the \( E_{[v]v}[v] \), \( E_{[q]v}[q] \), and \( E_{[h]v}[h] \). Finally, these four expectations are sent to the fully-connect layer, \( i.e. \), \( g(\cdot) \) in Figure 4, with the Softmax layer to calculate the answer distribution, \( P(A|do(V, Q, H)) \).

Given the video \( V \), the question \( Q \), the knowledge space \( Z \), the video feature space \( V_o, V_a, V_m \), the question feature space \( Q \), and the aligned feature space \( H \), the overall process is formulated as

\[
\hat{h}, \hat{q}, \hat{v}_o, \hat{v}_a, \hat{v}_m = \text{BaselineMethod}(\mathcal{V}, Q),
\]

\[
z = \text{EXP}(Z, \text{Cat}(\hat{h}, \hat{q})),
\]

\[
\hat{v}_o = \text{EXP}(V_o, \hat{v}_o),
\]

\[
\hat{v}_a = \text{EXP}(V_a, \hat{v}_a),
\]

\[
\hat{v}_m = \text{EXP}(V_m, \hat{v}_m),
\]

\[
\hat{q} = \text{EXP}(Q, \text{Cat}(\hat{v}_o, \hat{v}_a, \hat{v}_m)),
\]

\[
\hat{h} = \text{EXP}(H, \text{Cat}(\hat{v}_o, \hat{v}_a, \hat{v}_m, \hat{q})),
\]

\[
P(A|do(V, Q, H)) = \text{softmax}(g(z, \hat{v}_o, \hat{v}_a, \hat{v}_m, \hat{q}, \hat{h})).
\]

When training the KPI framework, we minimize the cross-entropy loss by using \( P(A^*|do(V, Q, H)) \) as the target, \( i.e. \), \( \mathcal{L} = -log P(A^*|do(V, Q, H)) \), where \( A^* \) indicates the ground-truth answer.

### 5. Experiments

#### 5.1. Experiment Settings

**Datasets.** We conduct experiments on five benchmark datasets that focus on the VideoQA from different aspects: MSVD-QA [65] and MSRVTT-QA [65] focus on the descriptive question, where QA pairs are automatically generated from the corresponding video captions datasets. TGIF-QA [24] splits the dataset into three subsets and emphasizes the action recognition, temporal state transition, and frame-level description, respectively. NExT-QA [61] features description, temporal relation, and evidence reasoning among multiple objects. Causal-VideoQA [32] challenges the reasoning ability from both evidence and commonsense. More details about the dataset statistics and implementation can be found in Supplementary Material.

**Baseline Methods.** Current VideoQA methods for video-question alignment can be divided into three categories, 1) the Memory-based methods that maintain a memory bank to boost the representation of video and question, \( e.g. \), CoMem [12], and HME [9]; 2) the Graph-based methods that exploit the graph networks to model the intra- and inter-relations between video and question, \( e.g. \), L-GCN [22], HGA [26], and B2A [40]; 3) the Hierarchy-based methods that study the multi-granularity natural hierarchy structure of video to enhance the video-question interaction, \( e.g. \), HCRN [28], MSPAN [16], HOSTR [7], and HQGA [62]; 4) the Video-Language pre-trained method that explore multiple task, like VideoQA, video caption, and
video-image retrieval to enhance video-language alignment in pre-trained models, e.g., VIOLET [11], JustAsk [68], and MERLOT [72]. To validate the generalization of our KPI framework, we migrate three baseline methods from different categories: CoMem [12] (memory), HGA [26] (graph), and HQGA [62] (hierarchy). Besides, we also compare with two causal VideoQA methods, IGV [35] and EIGV [34].

### 5.2. Main Results

In Table 1, we summarize the results of SOTA methods and those with our KPI framework on five benchmark datasets, i.e., MSVD-QA, MSRVTT-QA, TGIF-QA, NExT-QA, and Causal-VidQA. Note that we use the multi-head product attention. Con, Atomic, and both of them to filter correlated concepts.

<table>
<thead>
<tr>
<th>Model</th>
<th>MSVD-QA</th>
<th>MSRVTT-QA</th>
<th>NExT-QA</th>
<th>Causal-VidQA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Action</td>
<td>Transition</td>
<td>FrameQA</td>
<td></td>
</tr>
<tr>
<td>Memory-based</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CoMem [12]</td>
<td>34.7</td>
<td>35.1</td>
<td>70.3</td>
<td>76.6</td>
</tr>
<tr>
<td>HME [9]</td>
<td>33.7</td>
<td>33.0</td>
<td>73.9</td>
<td>77.8</td>
</tr>
<tr>
<td>Graph-based</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L-GCN [22]</td>
<td>34.3</td>
<td>33.7</td>
<td>74.3</td>
<td>81.1</td>
</tr>
<tr>
<td>HGA [26]</td>
<td>36.7</td>
<td>36.8</td>
<td>76.3</td>
<td>82.1</td>
</tr>
<tr>
<td>B2A [40]</td>
<td>37.2</td>
<td>36.9</td>
<td>75.9</td>
<td>82.6</td>
</tr>
<tr>
<td>Hierarchy-based</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HCRN [28]</td>
<td>36.1</td>
<td>35.6</td>
<td>75.1</td>
<td>81.2</td>
</tr>
<tr>
<td>MSPAN [16]</td>
<td>40.3</td>
<td>37.8</td>
<td>78.4</td>
<td>83.3</td>
</tr>
<tr>
<td>HOSTR [7]</td>
<td>39.4</td>
<td>35.9</td>
<td>75.6</td>
<td>83.0</td>
</tr>
<tr>
<td>HQGA [62]</td>
<td>41.2</td>
<td>38.6</td>
<td>76.9</td>
<td>85.6</td>
</tr>
<tr>
<td>Scene Bias</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IGV [35]</td>
<td>40.8</td>
<td>38.3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>EIGV [34]</td>
<td>42.6</td>
<td>39.3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CoMem + KPI</td>
<td>40.0</td>
<td>37.6</td>
<td>73.4</td>
<td>80.0</td>
</tr>
<tr>
<td>HGA + KPI</td>
<td>41.2</td>
<td>39.1</td>
<td>78.7</td>
<td>85.0</td>
</tr>
<tr>
<td>HQGA + KPI</td>
<td>43.3</td>
<td>40.0</td>
<td>79.3</td>
<td>88.3</td>
</tr>
</tbody>
</table>

Table 2. Comparison with VIOLET [11], JustAsk [68], and MERLOT [72] on MSVD-QA [65], MSRVTT-QA [66], NExT-QA [61], and Causal-VidQA [32]. *: reproduced with official code.

Table 3. Evaluation of the effectiveness of the EXP module and knowledge space. C-Att and P-Att indicate channel attention and product attention. Con, Atomic, and both of them to filter correlated concepts. * indicates the knowledge space of MSVD-QA and NExT-QA is mixed up. Best results are highlighted in bold.

<table>
<thead>
<tr>
<th>Setting</th>
<th>MSVD-QA</th>
<th>NExT-QA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HGA</td>
<td>HQQA</td>
</tr>
<tr>
<td>1</td>
<td>Baseline</td>
<td>36.7</td>
</tr>
<tr>
<td>2</td>
<td>KPI</td>
<td>41.2</td>
</tr>
<tr>
<td>3</td>
<td>EXP</td>
<td>C-Att</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>P-Att</td>
</tr>
<tr>
<td>5</td>
<td>Knowledge Space</td>
<td>Con</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>Atomic</td>
</tr>
<tr>
<td>7</td>
<td>*Con + Atomic</td>
<td>40.2</td>
</tr>
<tr>
<td>8</td>
<td>*Con</td>
<td>39.8</td>
</tr>
<tr>
<td>9</td>
<td>*Atomic</td>
<td>39.3</td>
</tr>
</tbody>
</table>

Table 1. Comparison with baseline methods on five datasets. Best results on each dataset are highlighted in bold. The improvement towards baseline method are highlight in red.

(1) On all benchmark datasets and for all baseline methods, our KPI framework outperforms the baseline methods by a large margin (+1.4%~+5.3%), which proves both the generalization ability and model-agnostic property of our KPI framework. As a derivation and extension of causal intervention, the distinct improvements further prove the effectiveness and generalization of front-door adjustment from both theoretical and empirical aspects.

(2) Comparing the improvements among different baseline methods, we notice that our KPI framework improves more on CoMem (2.5%~5.3%) and HGA (2.5%~4.5%) than HQGA (1.4%~3.8%). We suspect that the extra bounding-box feature and hierarchy interaction between video and question not only enhance the robustness of HQGA, but also help reduce the spurious correlations.

(3) Comparing the improvements among different benchmark datasets, we observe our KPI framework achieves the largest improvement on MSVD-QA (2.1%~5.3%) and achieves the smallest improvement on MSRVTT-QA (1.4%~2.5%). The reason for such observation is that MSRVTT-QA and MSVD-QA are the largest and smallest dataset. The baseline methods tend to capture the spurious correlations and overfit the training set with fewer training instances. Furthermore, we conjecture that KPI framework performs better in a less generalized situation, leading to the
improvement gap between MSVD-QA and MSRVTT-QA.

In Table 2, we further conduct experiment with video-language pre-trained models, where we can find that our KPI framework can boost the performance of the stronger baseline methods on these four datasets, which indicates the generalization ability of our framework in different situations. Compared among different baseline methods, our framework can achieve the most on NExT-QA and achieve the least on Causal-VidQA, which is about 1.3% - 1.9% and 0.6% - 0.7%, respectively. We suspect that the NExT-QA focuses more on evidence-based question-answering and the Causal-VidQA focuses more on commonsense-based question-answering, which make our deconfounding framework have less effect on the Causal-VidQA

5.3. Ablation study

In this section, we study the effects of different knowledge spaces, EXP modules, and hyper-parameters. All ablation experiments are conducted on MSVD-QA and NExT-QA with HGA and HQGA as baseline methods.

**The effect of different knowledge spaces.** We study the effect of knowledge space from two aspects in Table 3, *i.e.*, the difference between knowledge graphs and the difference between sharing and separating knowledge spaces between datasets. Comparing line 2 *v.s.* 5 *v.s.* 6 and line 7 *v.s.* 8 *v.s.* 9, we observe that jointly using both ConceptNet and Atomic is better than only using one of them. Besides, ConceptNet works better than Atomic for MSVD-QA, however, Atomic works better than ConceptNet for NExT-QA. This is because the relations on ConceptNet are mainly about physical entities, whereas the relations on Atomic emphasize more on event and social interaction. Therefore, ConceptNet is more helpful for description, and Atomic contributes more on reasoning. Comparing line 2 *v.s.* 7, we notice that the tendency between MSVD-QA and NExT-QA is inconsistent. For MSVD-QA, sharing the knowledge spaces undermines our KPI framework, but for NExT-QA, sharing the knowledge spaces further boosts the performance. We suspect that the MSVD-QA focuses on relatively simple scenes, which only requires limited knowledge for answer prediction. Hence, the knowledge from NExT-QA would introduce more noise than information. On the contrary, the NExT-QA focus on relatively complex scenes with temporal and evidence reasoning question, which requires more knowledge for answer prediction, and the knowledge space from MSVD-QA would be complementary.

**The effect of different EXP modules.** We validate the effect of different EXP modules in Table 3. Comparing the performance among lines 2 *v.s.* 3 *v.s.* 4, we find that product attention outperforms channel attention, while multi-head attention outperforms both of them. Regarding the difference between channel and product attention, product attention introduces 2nd-order interaction between key and query. In contrast, channel attention only exploits the 1st-order interaction, which helps product attention produce more informative attention weight. Furthermore, on top of the 2nd-order interaction, multi-head attention introduces more diverse attention weights, which could capture different attention patterns within a single attention layer.

**The effect of different variables for answer prediction.** We validate the effect of different variables for answer prediction in Table 4. Comparing the performance among lines 1 *v.s.* 2 *v.s.* 3 *v.s.* 4 *v.s.* 5 *v.s.* 6, we can find that the **H** contributes the most among **V**, **Q**, and **H** since **H** contains information from both **V** and **Q**. Moreover, we can also find that **Z** contributes the most among all four variables, since **Z** is in charge of the front-door adjustment, which enhance the generation ability of existing method.

**The effect of different variables for Z construction.** We validate the effect of different variables for Z construction in Table 4. Comparing the performance among lines 1 *v.s.* 2 *v.s.* 3 *v.s.* 4 *v.s.* 5 *v.s.* 6, we find that the **H** contributes the most among **V**, **Q**, and **H** since **H** contains information from both **V** and **Q**. Moreover, we can also find that **Z** contributes the most among all four variables, since **Z** is in charge of the front-door adjustment, which enhance the generation ability of existing method.

**The effect of dictionary size.** We change the **k_v**, **k_Q**, and **k_H** in the range of [100, 1000] with interval 100 in turn, and fix the size of the other two feature spaces as 500 to plot the performance variance in Figure 5. As feature space size in-
creases from 100 to 1000, the accuracy first increases and then becomes stable. We suspect that as the size of feature space increases, more base patterns in three feature spaces are introduced, which helps estimate the expectation in each space. However, as the feature size becomes larger, the speed of introducing new patterns becomes slower, which gradually stabilizes the accuracy curve.

5.4. Qualitative Results

To capture the learning insight of our KPI framework, we inspect the predictive answer of some video instances along with top-attended causal concepts and show the visualization in Figure 6. The causal concepts retrieved from our KPI framework provide comprehensive support for the answer prediction and effectively alleviate the effect of dataset bias. Besides, we also notice a few causal concepts may not be directly used for answer prediction, e.g., <hold-xReact-prevent escape>; but such causal concepts can reflect the characteristics of objects or actions, which could be helpful to exclude some wrong answers.

Moreover, we further show another two VideoQA example on MSVD-QA to reveal how our KPI framework reduce the bias. Our method can retrieve the knowledge items <pouch-AtLocation-kangaroo>, <dust-RelatedTo-spray>, and <target-RelatedTo-shoot>, which provides more causal clues to deduce the answer beyond correlation.

5.5. Limitation

As we have discussed in Section 4.1, the knowledge space should cover all the knowledge required for answer prediction, which is nearly impossible due to the following reasons, (1) existing knowledge graphs do not contain all the causal concepts required for answer prediction; (2) if the knowledge space size is too large, the resource required during training and inference will also be intolerable. Besides, current EXP module can only capture the 1st-order head-tail relation through the causal concepts. Nevertheless, more complex head-head-tail or head-tail-tail relations may also be helpful for answer prediction. In the future, exploring a more suitable knowledge space and designing a more informative EXP module would be the key to front-door adjustment in VideoQA task.

6. Conclusion

In this paper, we have focused on the VideoQA from dataset bias. Through analysis with causal graph, we have proven that the confounder and the backdoor path lead to spurious causality. Furthermore, we have proposed a model-agnostic framework called Knowledge Proxy Intervention, which has exploited the front-door adjustment and required no prior knowledge about the confounder. The effectiveness of KPI framework has been corroborated by three baseline methods on five benchmark datasets.

Acknowledgement

The work was supported by the National Natural Science Foundation of China (Grant No. 62076162), the Shanghai Municipal Science and Technology Major/Key Project, China (Grant No. 2021SHZDZX0102, Grant No. 20511100300). We thank Wu Wen Jun Honorary Doctoral Scholarship, AI Institute, Shanghai Jiao Tong University.
References


ing via gradually refined attention over appearance and motion. In *ACM MM 2017*, pages 1645–1653, 2017.  2, 6, 7, 9


