




# An Explainable Vision Question Answer Model via Diffusion Chain-of-Thought

Chunhao Lu<sup>1</sup>, Qiang Lu<sup>1</sup>, and Jake Luo<sup>2</sup>

<sup>1</sup> Beijing Key Laboratory of Petroleum Data Mining, China University of Petroleum  
Beijing, Beijing, China

`luchunhao@student.cup.edu.cn` `qianglu@cup.edu.cn`

<sup>2</sup> Department of Health Informatics and Administration, University of Wisconsin  
Milwaukee, Milwaukee, United States  
`jakeluo@uwm.edu`

**Abstract.** Explainable visual question-answering research focuses on generating explanations for answers. However, in complex VQA scenarios, there can be a significant semantic distance between the question and the answer. This means that generating explanations solely for the answer can lead to a semantic discrepancy between the content of the explanation and the question-answering content. To address this, we propose a step-by-step reasoning approach to reduce such semantic discrepancies. Additionally, the task of explaining VQA should include generating explanations for the reasoning steps to obtain explanations for the final answer. We introduce a diffusion chain-of-thought model to implement this step-by-step reasoning and the explanation process. The model consists of two processes: the external diffusion and the internal diffusion. The external diffusion process generates explanations for each reasoning step, while the internal diffusion process describes the probability of the question transitioning to each step of the explanation. Through experiments on eight sub-tasks in the ScienceQA dataset, we demonstrate that our diffusion chain-of-thought model outperforms GPT-3.5 in terms of the answer accuracy and the explanation ability while only using 1% of GPT-3.5’s parameters. Furthermore, the model approaches GPT-4, Llama, and so on in eight sub-tasks.

**Keywords:** Vision question answering · Explicable question answering  
· Chain-of-Thought · Diffusion model

## 1 Introduction

Vision Question Answering (VQA) aims to answer questions posed about a given image [2]. While traditional VQA methods can provide corresponding answers, they lack explanations for their outputs [10, 44, 47]. To address this, existing researches have explored three approaches: 1) Explanation Generation: This involves adding an interpretable module to the model to generate explanations for the answers [30, 45, 46, 51]. 2) Prototype Network: An interpretable QA model based on metric learning, which represents QA pairs as low-dimensional space



$Q, I \rightarrow R_1 \rightarrow R_2 \rightarrow \dots \rightarrow R_n$ ". The reasoning steps from " $Q, I$ " to " $R_n$ " can be interpreted as a diffusion process that conforms to the Markov chain [13, 53]. To achieve the process, this paper proposes an explainable VQA model based on the diffusion chain-of-thought (VQA Thought Diffusion, VQA-TD), as shown in Fig. 1. The model consists of three parts: the semantic embedding module, the semantic alignment module, and the reasoning module. The semantic embedding module uses the text model T5 [34] and the image model DETR [7] to implement the semantic embedding of the input  $Q$  and  $I$ . The semantic alignment module adopts the joint attention model Co-Attention [49] to fulfill the alignment of text and image semantics. The reasoning module consists of two diffusion processes: the external diffusion and the internal diffusion. Inspired by U-ViT [3] and DiT [31], both diffusion processes use the transformer decoder of the Markov diffusion process. They can obtain the transition probability ( $P(R_i \rightarrow R_{i+1})$ ) between reasoning explanations and the probability ( $P(Q, I \rightarrow R_i)$ ) of the question migrating to explanations. VQA-TD introduces the internal diffusion to enhance the relationship between the question and explanations, thereby reducing the semantic deviation between them.

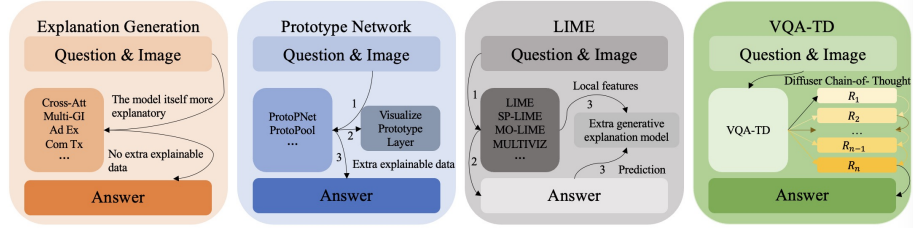
The main contributions are as follows:

- 1) We propose a novel VQA-TD neural network that addresses the limitations of existing VQA explanation models [5, 8, 12, 20, 23, 30, 35, 37, 41, 45, 46, 51]. Unlike traditional models that directly generate explanations, VQA-TD gradually obtains explanations for each reasoning step. It reduces the semantic gap between  $Q, I$  and answer. Different from the large language (multi-modal) models that use external CoT prompts to output explanations [11, 18, 21, 28, 32, 42, 43, 48, 52, 53], VQA-TD embeds the diffusion chain-of-thought model to produce explanations without external prompts.
- 2) We propose a diffusion chain-of-thought model composed of the external diffusion and the internal diffusion to support step-by-step explanations. It obtains the connection between reasoning steps through the external diffusion and generates the explanation for each reasoning according to the question through the internal diffusion.
- 3) Experiments on eight sub-tasks of ScienceQA dataset [26] demonstrate that VQA-TD, with only 1% of the parameter size of GPT-3.5 [6], surpasses GPT-3.5 in both the answer accuracy and the explanation capability. Also it approaches GPT-4 [1], Llama [39] and so on in the sub-tasks.

## 2 Related Work

Traditional VQA explainable methods include 1) Explanation Generation, 2) Prototype Network, and 3) LIME.

- 1) **Explanation Generation** has been proposed to improve the explainability of VQA by incorporating an interpretable module within the framework. The models leverage various techniques, including attention mechanisms, deep multi-modal reasoning, noise-resistant robustness design, and competitive explanation generation, as shown in Fig. 2.



**Fig. 2:** Comparison of VQA interpretable methods.

Wu *et al.* [46] proposed a VQA explanation generation that combines visual attention with an explanation generation module to capture the relationship between text and relevant image regions. The answer is then used to infer the corresponding relationship between these regions. Zhang *et al.* [51] proposed a deep multi-modal reasoning and fusion network. It performs fine-grained reasoning and adaptive fusion through multi-image reasoning and fusion layers. An explanation generation module is also designed to improve the explainability of answer. Patro *et al.* [30] proposed a robust explanation method. This method is resistant to noise perturbations. And it enhances the consistency between textual, visual explanations and the answer through a co-association module. Chen *et al.* [45] proposed a competitive textual explanation generation framework. It generates textual explanations for each answer by comparing the explanations of multiple competing answers. This framework aims to improve VQA performance on complex questions and enhance the interpretability of VQA systems.

2) **Prototype Network** [8, 12, 23, 37] is an interpretable QA model based on metric learning. It learns to represent QA as vectors in a low-dimensional space, and then calculates the distance between the QA (such as Euclidean distance or cosine similarity), as shown in Fig. 2.

Chen *et al.* [8] proposed the ProtoPNet model. It utilizes a generalized convolution as prototype layer to calculate the distance between QA. However, ProtoPNet needs to set up a separate prototype layer for each category, which makes its training process complex and the interpretability weak. To address this problem, Rymarczyk *et al.* [37] proposed the ProtoPool model. It significantly reduces the number of required prototype layers by sharing prototype layer between categories. In addition, ProtoPool introduces a new metric similarity function to help the model focus on more salient visual features to provide interpretability of the model.

3) **LIME** [5, 20, 35, 41] is an algorithm for explaining the predictions of black box models. It works by constructing a local linear model that approximates the relationship between local features and prediction outcome. LIME is implemented using a combination of techniques, including local interpretable model learning, representative predictive instance selection, optimized sampling and automated feature selection, and a structural visualization framework, as shown in Fig. 2.

Ribeiro *et al.* [35] proposed LIME for explaining the prediction decisions of classifiers. It learns an interpretable model to approximate the relationship between prediction and local features. Wang *et al.* [41] further proposed the MO-LIME based on LIME [35]. It improves efficiency and reduces manual intervention by optimizing the sampling process of predictive instance selection and automated feature selection. Liang *et al.* [20] proposed the multi-modal visualization model MULTIVIZ. It divides the explanation into four stages. In each stage, it provides existing and newly proposed analysis tools to identify the contribution of each modalities to the prediction, explore the interactions between modalities, reveal which feature combinations play an important role in model’s decision-making process, and analyze how decision-level features are combined to make predictions.

Although VQA explanation methods can provide explanations for the answer, they fail to provide a step-by-step explanation of the reasoning process. In complex scenarios, due to the long semantic distance of QA, those methods may fail to map. So this may cause semantic deviation. VQA-TD effectively avoids this problem by using diffusion models to explain each stage of reasoning. It ensures semantic distance short.

### 3 Diffusion Chain-of-Thought Model

The proposed diffusion chain-of-thought model incorporates two diffusion processes: the external diffusion and the internal diffusion. The external diffusion is used to obtain the transition probability in reasoning steps ( $P(R_i \rightarrow R_{i+1})$ ). The internal diffusion is used to obtain the probability of the question migrating to each explanation ( $P(< Q, I > \rightarrow R_i)$ ). It augments the internal diffusion to strengthen the relationship between the question and explanations, thereby reducing the semantic gap between them.

#### 3.1 Diffusion Model

Diffusion model is a generative model that learns to denoise data by gradually removing noise from a noisy input [13]. This process can be divided into two stages: a forward process and a reverse process.

In the forward process, the noise is gradually added to the real data. Given a real data point,  $n$  steps of Gaussian noise are cumulatively added to obtain  $n$  noisy data points. The noise at each step is generated from a Gaussian distribution controlled by a hyper-parameter  $\{\beta_i \in (0, 1)\}_{i=1}^n$ . Since each time step  $i$  only depends on the previous time step  $i - 1$ , the forward process can be considered as a Markov chain [13, 17] and is represented by the following two equations:

$$q(R_{1:n}|R_0) = \prod_{i=1}^n q(R_i|R_{i-1}) \quad (1)$$

$$q(R_i|R_{i-1}) = \mathcal{N}(R_i|\sqrt{\alpha_i}R_{i-1}, \beta_i\mathbf{I}) \quad (2)$$

among them, Eq. (1) is expressed as the cumulative noise migration process based on the Markov chain from  $R_0$  to  $R_{1:n}$ . Eq. (2) represents the noise addition relationship from  $R_{i-1}$  to  $R_i$ , where  $\beta_i$  is the  $i$ -th step Gaussian noise parameter,  $\alpha_i = 1 - \beta_i$ . It can be seen from Eq. (1) and Eq. (2) that  $R_i$  is formed by gradually adding Gaussian white noise  $\epsilon$  to the initial  $R_0$  [13].

$$R_i = \sqrt{\bar{\alpha}_i} R_0 + \sqrt{1 - \bar{\alpha}_i} \epsilon^R \quad (3)$$

where  $\bar{\alpha}_i = \prod_{j=1}^i \alpha_j$ .

The backward process is the process of gradually restoring the real data from Gaussian noise  $R_n \sim \mathcal{N}(0, \mathbf{I})$ , where each step of denoising is expressed as  $q(R_{i-1}|R_i)$ . Since the mathematical expression of  $q(R_{i-1}|R_i)$  cannot be obtained, the deep learning model  $p_\theta$  (parameter is  $\theta$ ) is used to estimate it [13, 17].

$$p_\theta(R_{0:n}) = p(R_n) \prod_{i=1}^n p_\theta(R_{i-1}|R_i) \quad (4)$$

$$p_\theta(R_{i-1}|R_i) = \mathcal{N}(R_{i-1}|\mu_\theta(R_i, i), \Sigma_\theta(R_i, i)) \quad (5)$$

where Eq. (4) is expressed as the cumulative denoising migration process based on the Markov chain from  $R_n$  to  $R_{0:n}$ . Eq. (5) represents the denoising relationship from  $R_i$  to  $R_{i-1}$ ,  $\mu_\theta(R_i, i)$  is the noise mean, and  $\Sigma_\theta(R_i, i)$  represents the noise variance [38]. The noise mean  $\mu_\theta$  can be expressed by the following Eq. (6).

$$\mu_\theta = \frac{1}{\sqrt{\alpha_i}} (R_i - \frac{\beta_i}{\sqrt{1 - \bar{\alpha}_i}} \epsilon_\theta(R_i, i)) \quad (6)$$

where  $\epsilon_\theta$  is the noise going from  $R_i$  at the  $i$  time.

### 3.2 External Diffusion

The external diffusion regards the intermediate reasoning steps from the question  $\langle Q, I \rangle$  to the answer  $R_n$  ( $\langle Q, I \rangle \Rightarrow R_1 \rightarrow R_2 \rightarrow \dots \rightarrow R_n$ ) as a diffusion process. This process is used to describe the transition probability  $P(R_i \rightarrow R_{i+1})$  in reasoning steps, where  $\langle Q, I \rangle$  is the initial state  $R_0$  of the process. And the intermediate reasoning step  $R_i$  is injected with Gaussian noise by  $R_{i-1}$ .  $R_n$  represents the final answer state. This process (shown in Fig. 1(a)) can be expressed by Eq. (1) and Eq. (2).

### 3.3 Internal Diffusion

In the process of adding noise to the internal diffusion (as shown in Fig. 1(d)), the initial question  $f_0$  is  $\langle Q, I \rangle$ , and  $r_0$  is composed of two consecutive interpretation steps ( $R_i \oplus R_{i-1}$ ) in the external diffusion process.  $r_t$  is generated by adding noise to  $r_{t-1}$ , which can be expressed by Eq. (2).  $f_t$  is the content formed by introducing noise under the two conditions of  $f_{t-1}$  and  $r_t$ , and the relationship between them is expressed by the full probability  $q(f_t, f_{t-1}, r_t)$ .

In the denoising and restoration process of the internal diffusion shown in Fig. 1(e),  $f_{t-1}$  is obtained by denoising on  $f_t$ , which is expressed by Eq. (5). In addition,  $r_t$  is obtained by denoising on  $f_t$  and  $r_{t+1}$ . It is formed by denoising reduction under two conditions, and the relationship between them is represented by the total probability  $p_\theta(r_t, r_{t+1}, f_t)$ .

According to [14], the total probability model is

$$p_\theta(r_t, r_{t+1}, f_t) = (1 + s)p_\theta(f_t, r_t) - s[p_\theta(f_t|f_{t+1}), p_\theta(r_t|r_{t+1})]. \quad (7)$$

Based on Theorem 1,  $p_\theta(r_t, r_{t+1}, f_t)$  is approximately equal to the full probability noise distribution model  $\hat{\epsilon}_\theta(f_t, r_t, t)$  that is computed by Eq. (8).

**Theorem 1.** Assuming that  $p_\theta(f_t, r_t)$ ,  $p_\theta(f_t|f_{t+1})$  and  $p_\theta(r_t|r_{t+1})$  follow the noise boundary distribution models  $\epsilon_\theta(f_t, r_t, t, t)$ ,  $\epsilon_\theta^f(f_t, \epsilon^r, t, n)$  and  $\epsilon_\theta^r(\epsilon^f, r_t, n, t)$  respectively.  $p_\theta(r_t, r_{t+1}, f_t) \approx \hat{\epsilon}_\theta(f_t, r_t, t)$ , where

$$\hat{\epsilon}_\theta(f_t, r_t, t) = (1 + s)\epsilon_\theta(f_t, r_t, t, t) - s \begin{bmatrix} \epsilon_\theta^f(f_t, \epsilon^r, t, n), \\ \epsilon_\theta^r(\epsilon^f, r_t, n, t) \end{bmatrix} \quad (8)$$

*Proof.* The proof is in the Appendix A.

According to [4] and Bayesian formula, there is an approximate relationship in denoising score matching loss:

$$\epsilon_\theta(f_t, r_t, t, t) \approx -\sqrt{\bar{\beta}_t} \begin{bmatrix} \nabla_{f_t} \log q(f_{t-1}|f_t, r_0) \\ \nabla_{r_t} \log q(r_{t-1}|r_t, f_0) \end{bmatrix}. \quad (9)$$

Based on Theorem 2,  $\hat{\epsilon}_\theta(f_t, r_t, t)$  is approximately equal to denoising score matching loss that be computed by Eq. (10).

**Theorem 2.** According to Eq. (8), assuming noise model  $\epsilon_\theta(f_t, r_t, t, t)$ ,  $\epsilon_\theta^f(f_t, \epsilon^r, t, n)$  and  $\epsilon_\theta^r(\epsilon^f, r_t, n, t)$  follow the denoising score matching loss and the classifier-free guidance. So the  $\hat{\epsilon}_\theta(f_t, r_t, t)$  can be approximated by

$$\hat{\epsilon}_\theta(f_t, r_t, t) \approx -\sqrt{\bar{\beta}_t} \begin{bmatrix} (1 + s)\nabla_{f_t} \log q(f_{t-1}|f_t, r_0) - s\nabla_{f_t} \log q(f_t), \\ (1 + s)\nabla_{r_t} \log q(r_{t-1}|r_t, f_0) - s\nabla_{r_t} \log q(r_t) \end{bmatrix} \quad (10)$$

where  $\bar{\beta}_t$  is equal to  $\prod_{i=1}^n \beta_t$  which stands for hyper-parameter of Gaussian distribution,  $q(f_{t-1}|f_t, r_0)$  is conditional probability distributions based on the question and two consecutive explanation steps respectively, and  $q(f_t)$  is boundary probability distributions standing for the state of question  $f$ .

*Proof.* the proof is shown in the Appendix A.

## 4 VQA-TD

### 4.1 Model Architecture

The VQA-TD algorithm model framework is divided into three main modules: the semantic embedding, the semantic alignment, and the reasoning module (as

shown in Fig. 1). The semantic embedding module uses T5 [34] and DETR [7] to perform semantic embeddings of  $Q$  and  $I$ . The semantic alignment module uses the joint attention model Co-Attention [49] to semantically align  $Q$  and  $I$ . Inspired by U-ViT [3] and DiT [31], the reasoning module uses a transformer decoder that conforms to the Markov diffusion process. Then it gradually restores explanations of the question to the answer. It restores the explanation of  $M : < Q, I > \rightarrow R_1 \rightarrow R_2 \rightarrow \dots \rightarrow R_n$  in  $R_1 R_2 \dots R_n$ . And it contains two diffusion processes: the external diffusion and the internal diffusion. The external diffusion process is used to describe the logical probability relationship in reasoning explanations. The internal diffusion process uses the intermediate explanation  $R_i$  for the noise generation and the restoration of  $< Q, I >$ . Therefore, it enables learning the probability distribution of the question to each step explanation  $R_i$ .

#### 4.2 Semantic Embedding Module

The semantic embedding module is used to embed  $< Q, I >$ . Since  $Q$  and  $I$  are different modal data, this embedding module uses the T5 model [34] and DETR model [7] to handle  $Q$  and  $I$ , respectively. It achieves the embedding of two modal data.  $Q$  includes the question, the prompt data, and the option text content, while  $I$  contains the image data about the question itself.

During the semantic embedding process, two types of data can be expressed as:

$$H_l = \text{LanguageEncoder}(Q) \quad (11)$$

$$H_v = W_h \cdot \text{VisionEncoder}(Q) \quad (12)$$

in order to unify the lengths of two latter semantic vectors of embedding, this method constructs a projection matrix  $W_h$  to convert the image visual semantic embedding vector into a language embedding vector of equal length.

#### 4.3 Semantic Alignment Module

To align the multi-modal data in VQA, this paper adopts the alignment attention mechanism designed by Yu *et al.* [49] and divides the alignment process into two parts: 1) self-attention unit(SA). It simulates the internal semantic connections of each modality. It learns coarse-grained semantic connections, allowing the model to understand the internal semantic structure of the modality deeply. 2) guided-attention(GA) unit. It solves the semantic alignment problem between modalities. This process helps the model understand the relationship between image and questions more accurately.

In the SA unit, the query vector  $Q$ , the key vector  $K$ , and the value vector  $V$  correspond to the language features  $H_l$ , visual features  $H_v$ , and  $H_v$ , respectively. So the output  $H_v^{att}$  of the SA unit can be defined as:

$$H_v^{att} = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \quad (13)$$



Algorithm 1 Forward noise adding process	Algorithm 2 Reverse denoising process
1: repeat	1: $f_n, r_n \sim \mathcal{N}(0, \mathbf{I})$
2: $f_0, r_0 \sim q(f_0, r_0)$	2: for $t = n, \dots, 1$ do
3: $t \sim \text{Uniform}(\{1, 2, \dots, n\})$	3: $z^f, z^r \sim \mathcal{N}(0, \mathbf{I})$ if $t \geq 1$ ,
4: $\epsilon^f, \epsilon^r \sim \mathcal{N}(0, \mathbf{I})$	else $z^f, z^r = 0$
5: $f_t = \sqrt{\alpha_t} f_0 + \sqrt{1 - \alpha_t} \epsilon^f$	4: $f_{t-1} = \frac{1}{\sqrt{\alpha_t}} (f_t - \frac{\beta_t}{\sqrt{\alpha_t}} \epsilon_\theta^f(f_t, r_t, t)) + \sigma_t z^f$
6: $r_t = \sqrt{\alpha_t} r_0 + \sqrt{1 - \alpha_t} \epsilon^r$	5: $r_{t-1} = \frac{1}{\sqrt{\alpha_t}} (r_t - \frac{\beta_t}{\sqrt{\alpha_t}} \epsilon_\theta^r(f_t, r_t, t)) + \sigma_t z^r$
7: Computing gradient descent via BP: $\nabla_\theta \ [\epsilon^f, \epsilon^r] - \hat{\epsilon}_\theta(f_t, r_t, t)\ _2^2$	6: end for
8: until converged	7: return $f_0, r_0$

the dimension size of  $d_k$  is equal to the dimension size of  $H_l$ .

Since only one head is used in the SA unit to simulate the dense interaction between modalities, the relationship between the words in the question and the image area cannot be well mapped. Therefore, this method uses the GA unit to perform cross-fusion of modalities to generate the fusion vector  $f$ :

$$\lambda = \text{Sigmoid}(W_l H_l + W_v H_v^{att}) \quad (14)$$

$$f = (1 - \lambda) H_l + \lambda \cdot H_v^{att} \quad (15)$$

where  $W_l$  and  $W_v$  are pre-trained learning parameters. This method fine-tunes the pre-training parameters in small dimensions to complete the encoding of  $f$ .

Finally, the conversion and generation of the deep latent space embedding vector can be completed by connecting  $f$ ,  $r$ , and their corresponding sampling time steps  $t$  together.

#### 4.4 Reasoning Module

The reasoning module is used to diffuse the chain of thought. It uses the external diffusion process to describe the transition probability relationship in reasoning steps so that there is a logical progression ability between the previous and subsequent reasoning steps. In addition, it uses the internal diffusion process to generate each reasoning explanation of the steps (as shown in Fig. 1(a)(d)(e)).

In order to fulfill the internal diffusion reduction process, this paper constructs VQA-TD based on transformer, as shown in Fig. 1(c). It consists of 1) Noised Latent Layer, 2) Embed Layer, 3) Patchify Layer, 4) Transformer Block, 5) Layer Normalization (LN), and 6) Linear Layer. Among them, the noised latent layer, the embed layer and the patchify layer are all fully connected networks. They are used to reduce the dimensionality of the aligned semantic vectors. They also introduce the condition information (inference time  $t$  and continuous inference interpretation step  $r = R_i \oplus R_{i-1}$ ) and output semantic vector tokens  $f$ . The transformer block layer is used to calculate the optimal noise  $\hat{\epsilon}_\theta(f_t, r_t, t)$  and is required to predict the reduction process of VQA-TD in

internal diffusion. LN and linear layer are used to initialize layer operations. Finally, the semantic vector  $f$  and the continuous interpretation step  $r$  are restored from tokens to the original input dimensions through LN and linear layer.

Transformer block utilizes the multi-head attention mechanism in transformer [40] to associate the semantic vector  $f$  with the explanation  $r$  in  $q(f_0, r_0)$  (line 2 in Algorithm 1). In addition, transformer block adds Multi-Layer perceptron (MLP), Scale, and Shift layers. Among them, MLP is used for spatial mapping. The scale is used for spatial alignment. The shift is used to denoise the continuous interpretation  $r$  and the semantic vector  $f$  respectively by  $\epsilon^f$  and  $\epsilon^r$  (lines 4-6 in Algorithm 1). The L2 loss relationship between the noise  $\hat{\epsilon}_\theta(f_t, r_t, t)$  obtained by the diffusion chain-of-thought model through the calculation process of Eq. (10), and the standard noise  $\epsilon^f$  and  $\epsilon^r$  that actually act on the embedded semantics  $f$  and  $r$  can be defined as the training convergence target (line 7 in Algorithm 1).

$$\mathbb{E}_{f_0, r_0, \epsilon^f, \epsilon^r, t} \|\epsilon^f, \epsilon^r\| - \hat{\epsilon}_\theta(f_t, r_t, t)\|_2^2 \quad (16)$$

LN performs zero initialization processing on the transmitted semantic vector  $f$  and the residual block inside the transformer block. LN is between the transformer block and the linear layer. In addition, the linear layer acts as a decoder to restore the input semantic vector  $f$  and continuously interpret  $r$  to the original dimension. That is, each token in the semantic vector is first mapped to the tensor space, and the reshape operation is performed simultaneously on the linear layer. The final feature dimension size is twice the original size, which is used to predict noise and variance in the restoration process.

In order to realize the internal diffusion reduction process, the reasoning module takes the continuous inference step  $r$  as the condition and reducing  $f$  to explain  $R_i$ . After completing the training of the total noise model  $\hat{\epsilon}_\theta$  process in Eq. (16), VQA-TD will use the total probability noise model  $\hat{\epsilon}_\theta(f_t, r_t, t)$ . It obtains  $\hat{\epsilon}_\theta$  in the forward diffusion and the reverse reduction processes. In addition, the noise mean  $\mu_\theta$  and variance  $\sigma_t$  in both processes corresponding to  $\epsilon_\theta^f(f_t, r_t, t, t)$  and  $\epsilon_\theta^r(f_t, r_t, t, t)$  (as shown in Algorithm 2 lines 4-5) are restored on  $f$  and  $r$  respectively. That is, through the correlation calculation relationship between the noise mean  $\mu_\theta$  and the variance  $\sigma_t$  in Appendix A, the Gaussian white noise is cyclically denoised in lines 4-5 of Algorithm 2.

## 5 Experiment and Result Analysis

### 5.1 Settings

This paper compares VQA-TD with the following 9 benchmark methods on ScienceQA [26], including: VisualBERT [19], UnifiedQA<sub>Base</sub> [16], UnifiedQA<sub>Base</sub> w/CoT [26], GPT-3.5 [9], GPT-3.5 w/CoT [26], GPT-4 w/CoT [27], Multi-Modal CoT Base(MC<sub>Base</sub>) [26], LLaMA-Adapter [50] and LLaMA-SciTune [15].

ScienceQA is a data set of scientific QA. It consists of 21,208 multiple-choice questions, covering rich domain diversity in 3 disciplines, 26 topics, 127 categories, and 379 skills. The benchmark data set is divided into the training set,

**Table 1:** Comparison accuracy test results % (NAT: natural science; SOC: social science; LAN: language science; TXT: text prompt; IMG: image prompt; NO: no prompt; G1-6: questions for grades 1-6; G7-12: questions for grade 7-12)

Model	Size	NAT	SOC	LAN	TXT	IMG	NO	G1-6	G7-12	Avg
Human	-	<b>90.2</b>	85.0	87.5	89.6	<b>87.5</b>	88.1	<b>91.6</b>	82.4	88.4
VisualBERT [19]	111M	59.3	69.2	62.2	62.7	62.2	58.5	63.0	59.9	61.9
UnifiedQA <sub>Base</sub> [16]	223M	68.2	69.2	74.9	63.8	61.4	77.8	73.0	65.0	70.1
UnifiedQA <sub>Base</sub> w/CoT [26]	223M	71.0	76.0	78.9	66.4	66.5	81.8	77.1	68.8	74.1
GPT-3.5 [9]	175B	74.6	69.7	76.0	74.4	67.3	77.4	76.8	68.9	74.0
GPT-3.5 w/CoT [26]	175B	75.4	70.9	78.1	74.7	67.4	79.9	78.2	69.7	75.2
GPT-4 w/CoT [27]	-	85.5	72.4	<b>90.3</b>	82.6	71.5	<b>92.9</b>	86.7	79.0	84.0
MC <sub>Base</sub> [26]	223M	87.5	77.2	85.8	87.9	82.9	86.8	84.7	85.4	84.9
LLaMA-Adapter [50]	6B	84.4	88.3	84.4	83.7	80.3	86.9	85.8	84.1	85.2
LLaMA-SciTune [15]	13B	89.3	<b>95.6</b>	87.0	<b>93.1</b>	86.7	91.8	84.4	<b>91.3</b>	<b>90.0</b>
<b>VQA-TD</b>	583M	83.2	71.8	81.3	82.4	78.2	80.6	79.1	80.5	79.7

validation set, and test set, containing 12726, 4241, and 4241 samples, respectively. These questions, answer prompts and multipart explanations serve as reasoning steps and explanations from question to answer.

In the process of training the model, the total training network layer is controlled to 26 layers (the number of attention mechanism heads is 18), which includes 4 transformer blocks, LN and linear layers. In addition, in the patch operation, this paper sets its parameter to 2 to determine the number of token magnitudes of the fusion vector  $f$ .

This paper completes the internal diffusion calculation process of diffusion chain-of-thought of Eq. (10) by setting the time steps of  $f$  and  $r$  to both  $t$ . During the training phase, VQA-TD follows the multi-stage pattern of steady-state diffusion [36]: in the first stage, 250K steps are trained at  $256 \times 256$  resolution with a batch size of 4096, and 5K steps of warm-up are performed. In the second stage, the model is fine-tuned for 200K steps at  $512 \times 512$  resolution with a batch size of 1024 and warmed up for 5K steps. At the end of the second phase, this paper recovers from its last checkpoint (including the model’s weights and the optimizer’s state). Following by the method of Bao *et al.* [4], this paper uses the AdamW optimizer [24] in all stages, with a learning rate of  $2e-4$ , a weight attenuation of 0.03, and an operating coefficient of  $(\beta_1, \beta_2)=(0.9, 0.9)$ . When the validation loss does not decrease, the training process will reduce the learning rate by 10 times and then continue training. Therefore, this paper uses mixed precision training. That is, when VQA-TD is trained at  $256 \times 256$  resolution, the image-related position embeddings are interpolated via bi-linear interpolation. During training, this paper uses the DPM-Solver [25] method to accelerate the diffusion process every 50 steps.

In order to verify the quality of the explanation text generated by each algorithm model, this paper uses the text BLEU [29] and ROUGE [22] evaluation

indicators as testing tools to evaluate the explanation output of each model. Finally, BLEU-1, BLEU-4, and ROUGE-L are used as test baselines, respectively.

## 5.2 Result Analysis

**Comparative Results** VQA-TD and the other four groups of models are used to complete eight sub-tasks in the ScienceQA data set. The specific results are shown in Tab. 1. Among them, the first group is the traditional VQA interpretation baseline model VisualBERT [19]. The second group includes the text-to-text language model UnifiedQA<sub>Base</sub> [16], and the language model after adding CoT UnifiedQA<sub>Base</sub> w/CoT [26]. The third group is GPT-3.5 [26]. The fourth group is LMMs, including GPT-4 [27], MC<sub>Base</sub> [26], LLaMA-Adapter [50] and LLaMA-SciTune [15].

With 1% of the parameters of GPT-3.5, VQA-TD (79.7%) surpasses VisualBERT (61.9%), UnifiedQA (74.1%) and GPT-3.5 (75.2%) respectively; with LLaMA-SciTune’s 5% parameter, VQA-TD is close to LMMs such as GPT-4 (84.0%), as shown in Tab. 1 and Fig. 1(b).

VQA-TD performs best on the NAT sub-task, with a rate of 83.2%. And it performs worst on the SOC sub-task, with a rate of 71.8%. Due to most of the questions in the NAT have text and image prompts, which are related to reasoning steps and explanations, this makes VQA-TD effectively capture information through the internal and external diffusion process. The lack of text prompt information in the SOC reduces VQA-TD’s ability to establish the connection between images and explanation. It can be seen that the inclusion of valid text and image data information in the data set is crucial to VQA-TD.

**Explain Answer Recall** This experiment uses BLEU [29] and ROUGE [22] to test the recall rate of explanations. VQA-TD, with 583M parameters (1% GPT-3.5), exceeds BLEU-1 of UnifiedQA<sub>Base</sub> (0.397), GPT-3.5 w/CoT (0.192) and MC<sub>Base</sub> (0.406) respectively by getting score of 0.421. It also exceeds BLEU-4 of UnifiedQA<sub>Base</sub> (0.370), GPT-3.5 w/CoT (0.052) and MC<sub>Base</sub> (0.384) respectively by getting score of 0.407. And it surpasses ROUGE-L of UnifiedQA<sub>Base</sub> (0.714), GPT-3.5 w/CoT (0.323) and MC<sub>Base</sub> (0.769) respectively by getting score of 0.820. Besides, it is also close to GPT-4 w/CoT (0.839) and LLaMA-Adapter (0.868) in ROUGE-L, as shown in Tab. 2. Although VQA-TD (79.7%) is worse than MC<sub>Base</sub> in accuracy (84.9%), it (BLEU-1: 0.421; BLEU-4: 0.407; ROUGE-L: 0.820) performs better than MC<sub>Base</sub> (BLEU-1: 0.406; BLEU-4: 0.384; ROUGE-L: 0.769) in BLEU and ROUGE. Due to VQA-TD have the ability to generate long explanation information. It gradually approximates long explanation information through multiple diffusion and recovery steps, thereby reducing the semantic deviation between  $\langle Q, I \rangle$  and the answer.

Compared with LMMs such as GPT-4 w/CoT and LLaMA-Adapter, VQA-TD performs worse on BLEU-1/4 (BLEU-1: 0.421; BLEU-4: 0.407). Since the model uses parameters in the multi-modal data alignment process. Co-Attention

**Table 2:** Recall comparison

Model	BLEU-1	BLEU-4	ROUGE-L
UnifiedQA <sub>Base</sub>	0.397	0.370	0.714
GPT-3.5 w/CoT	0.192	0.052	0.323
GPT-4 w/CoT	0.927	<b>0.894</b>	0.839
MC <sub>Base</sub>	0.406	0.384	0.769
LLaMA-Adapter	<b>0.944</b>	0.875	<b>0.868</b>
<b>VQA-TD</b>	0.421	0.407	0.820

**Table 3:** Different prompt recalls

Prompt Type	BLEU-1	BLEU-4	ROUGE-L
Img prompt	0.412	0.390	0.779
Txt prompt	0.417	0.393	0.781
No prompt	0.408	0.388	0.759
Both prompts	<b>0.421</b>	<b>0.407</b>	<b>0.820</b>

**Table 4:** Different prompt accuracy

Prompt Type	Accuracy(%)
Img prompt	77.2
Txt prompt	75.6
No prompt	73.3
Both prompts	<b>79.7</b>

[49], rather than a model with a larger parameter and richer semantic relationships between modalities such as CLIP [33]. This leads to poor semantic correlation between different modal data and large numbers of semantic errors.

**Ablation** This paper conducts experiments on accuracy and recall rates in the presence of four prompt data. Four prompt data are as follows: image prompt, text prompt, no prompt, and both prompts. VQA-TD is tested on the four different prompt data on interpretation recall. The experimental results show that both types of prompts are significantly better than the other three prompts (as shown in Tab. 3). Because prompts are related to reasoning steps and explanations, and richer prompt data types will make VQA-TD’s internal and external diffusion process more effective in capturing information. The results of providing only image prompts and only text prompts show that they have almost the same impact on the VQA-TD algorithm model.

VQA-TD is also tested on four sets of different prompt data in terms of answer accuracy. The results are similar to those in the previous set of tests. That is, the accuracy obtained by both prompt types is significantly better than the other three types (as shown in Tab. 4). However, the gap between "image prompt" and "text prompt" is still relatively large. So, the lack of image prompts will cause the accuracy to decline faster. Compared with text data, the internal diffusion process can more easily capture the semantic information in the image.

### 5.3 Limitations and improvements

VQA-TD achieves good results in multi-modal tasks, and the explanation text is robust. However, VQA-TD also has certain shortcomings. For example, inac-



## Acknowledgements

This work is supported by China National Key Research Project (No.2019YFC0312003).

## References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: Vqa: Visual question answering. In: Proceedings of the IEEE international conference on computer vision. pp. 2425–2433 (2015)
3. Bao, F., Li, C., Cao, Y., Zhu, J.: All are worth words: a vit backbone for score-based diffusion models. arXiv preprint arXiv:2209.12152 (2022)
4. Bao, F., Li, C., Zhu, J., Zhang, B.: Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. arXiv preprint arXiv:2201.06503 (2022)
5. Boukhers, Z., Hartmann, T., Jürjens, J.: Coin: Counterfactual image generation for vqa interpretation. arXiv preprint arXiv:2201.03342 (2022)
6. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. *Advances in neural information processing systems* **33**, 1877–1901 (2020)
7. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
8. Chen, C., Li, O., Tao, D., Barnett, A., Rudin, C., Su, J.K.: This looks like that: deep learning for interpretable image recognition. *Advances in neural information processing systems* **32** (2019)
9. Chen, T., Kornblith, S., Swersky, K., Norouzi, M., Hinton, G.E.: Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems* **33**, 22243–22255 (2020)
10. Chun, S., Oh, S.J., De Rezende, R.S., Kalantidis, Y., Larlus, D.: Probabilistic embeddings for cross-modal retrieval. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8415–8424 (2021)
11. Dai, D., Dong, L., Hao, Y., Sui, Z., Chang, B., Wei, F.: Knowledge neurons in pretrained transformers. arXiv preprint arXiv:2104.08696 (2022)
12. Hayes, T.L., Kafle, K., Shrestha, R., Acharya, M., Kanan, C.: Remind your neural network to prevent catastrophic forgetting. In: European Conference on Computer Vision. pp. 466–483. Springer (2020)
13. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. *Advances in neural information processing systems* **33**, 6840–6851 (2020)
14. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
15. Horawalavithana, S., Munikoti, S., Stewart, I., Kvinge, H.: Scitune: Aligning large language models with scientific multimodal instructions. arXiv preprint arXiv:2307.01139 (2023)
16. Khashabi, D., Min, S., Khot, T., Sabharwal, A., Tafjord, O., Clark, P., Hajishirzi, H.: Unifiedqa: Crossing format boundaries with a single qa system. arXiv preprint arXiv:2005.00700 (2020)

17. Kingma, D., Salimans, T., Poole, B., Ho, J.: Variational diffusion models. *Advances in neural information processing systems* **34**, 21696–21707 (2021)
18. Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y.: Large language models are zero-shot reasoners. *Advances in neural information processing systems* **35**, 22199–22213 (2022)
19. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557* (2019)
20. Liang, P.P., Lyu, Y., Chhablani, G., Jain, N., Deng, Z., Wang, X., Morency, L.P., Salakhutdinov, R.: Multiviz: Towards visualizing and understanding multimodal models. In: *The Eleventh International Conference on Learning Representations* (2022)
21. Lin, B.Y., Chen, X., Chen, J., Ren, X.: Kagnet: Knowledge-aware graph networks for commonsense reasoning. *arXiv preprint arXiv:1909.02151* (2019)
22. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: *Text summarization branches out*. pp. 74–81 (2004)
23. Liu, X., Ji, Z., Pang, Y., Han, J., Li, X.: Dgig-net: Dynamic graph-in-graph networks for few-shot human–object interaction. *IEEE Transactions on Cybernetics* **52**(8), 7852–7864 (2021)
24. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017)
25. Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., Zhu, J.: Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems* **35**, 5775–5787 (2022)
26. Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.W., Zhu, S.C., Tafjord, O., Clark, P., Kalyan, A.: Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems* **35**, 2507–2521 (2022)
27. Lu, P., Peng, B., Cheng, H., Galley, M., Chang, K.W., Wu, Y.N., Zhu, S.C., Gao, J.: Chameleon: Plug-and-play compositional reasoning with large language models. *Advances in Neural Information Processing Systems* **36** (2024)
28. Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J., Wu, X.: Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering* (2024)
29. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. pp. 311–318 (2002)
30. Patro, B., Patel, S., Namboodiri, V.: Robust explanations for visual question answering. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. pp. 1577–1586 (2020)
31. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 4195–4205 (2023)
32. Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A.H., Riedel, S.: Language models as knowledge bases? *arXiv preprint arXiv:1909.01066* (2019)
33. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: *International conference on machine learning*. pp. 8748–8763. PMLR (2021)



34. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* **21**(1), 5485–5551 (2020)
35. Ribeiro, M.T., Singh, S., Guestrin, C.: " why should i trust you?" explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. pp. 1135–1144 (2016)
36. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 10684–10695 (2022)
37. Rymarczyk, D., Struski, Ł., Górszczak, M., Lewandowska, K., Tabor, J., Zieliński, B.: Interpretable image classification with differentiable prototypes assignment. In: *European Conference on Computer Vision*. pp. 351–368. Springer (2022)
38. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: *International conference on machine learning*. pp. 2256–2265. PMLR (2015)
39. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023)
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems* **30** (2017)
41. Wang, B., Pei, W., Xue, B., Zhang, M.: A multi-objective genetic algorithm to evolving local interpretable model-agnostic explanations for deep neural networks in image classification. *IEEE Transactions on Evolutionary Computation* (2022)
42. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems* **35**, 24824–24837 (2022)
43. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems* **35**, 24824–24837 (2022)
44. Whitehead, S., Wu, H., Ji, H., Feris, R., Saenko, K.: Separating skills and concepts for novel visual question answering. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5632–5641 (2021)
45. Wu, J., Chen, L., Mooney, R.J.: Improving vqa and its explanations by comparing competing explanations. *arXiv preprint arXiv:2006.15631* (2020)
46. Wu, J., Mooney, R.J.: Faithful multimodal explanation for visual question answering. *arXiv preprint arXiv:1809.02805* (2018)
47. Xia, Q., Yu, C., Hou, Y., Peng, P., Zheng, Z., Chen, W.: Multi-modal alignment of visual question answering based on multi-hop attention mechanism. *Electronics* **11**(11), 1778 (2022)
48. Yasunaga, M., Ren, H., Bosselut, A., Liang, P., Leskovec, J.: Qa-gnn: Reasoning with language models and knowledge graphs for question answering. *arXiv preprint arXiv:2104.06378* (2021)
49. Yu, Z., Yu, J., Cui, Y., Tao, D., Tian, Q.: Deep modular co-attention networks for visual question answering. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 6281–6290 (2019)
50. Zhang, R., Han, J., Zhou, A., Hu, X., Yan, S., Lu, P., Li, H., Gao, P., Qiao, Y.: Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199* (2023)

51. Zhang, W., Yu, J., Zhao, W., Ran, C.: Dmrfnet: deep multimodal reasoning and fusion for visual question answering and explanation generation. *Information Fusion* **72**, 70–79 (2021)
52. Zhang, X., Bosselut, A., Yasunaga, M., Ren, H., Liang, P., Manning, C.D., Leskovec, J.: Greaselm: Graph reasoning enhanced language models. In: *International conference on learning representations* (2021)
53. Zhang, Z., Zhang, A., Li, M., Smola, A.: Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493* (2022)