Graph-Guided Textual Explanation Generation Framework

Anonymous ACL submission

Abstract

Natural language explanations (NLEs) are commonly used to provide plausible free-text explanations of a model's reasoning about its predictions. However, recent work has questioned their faithfulness, as they may not accurately reflect the model's internal reasoning process regarding its predicted answer. In contrast, highlight explanations–input fragments critical for the model's predicted answers–exhibit measurable faithfulness. Building on this foundation, we propose **G-Tex**, a **G**raph-Guided **T**extual **Ex**planation Generation framework designed to enhance the faithfulness of NLEs. Specifically, highlight explanations are first extracted as faithful cues reflecting the model's reasoning logic toward answer prediction. They are subsequently encoded through a graph neural network layer to guide the NLE generation, which aligns the generated explanations with the model's underlying reasoning toward the predicted answer. Experiments on T5 and BART using three reasoning datasets show that G-Tex improves NLE faithfulness by up to 12.18% compared to baseline methods. Additionally, G-Tex generates NLEs with greater semantic and lexical similarity to human-written ones. Human evaluations show that G-Tex can decrease redundant content and enhance the overall quality of NLEs. Our work presents a novel method for explicitly guiding NLE generation to enhance faithfulness, serving as a foundation for addressing broader criteria in NLE and generated text.

1 Introduction

Natural Language Explanations (NLEs) produce human-understandable texts to explain the model's prediction process (Wiegreffe et al., 2021). Self-rationalization, where the prediction and the corresponding NLE are generated simultaneously, is a commonly used method for NLE generation, which leads to improved agreement between the generated NLE and the produced prediction (Alvarez Melis and Jaakkola, 2018; Marasovic et al., 2022). However, existing work (Kumar and Talukdar, 2020; Wiegreffe et al., 2021) has found that these *NLEs are often unfaithful*, as they may present misleading reasons unrelated to the model's true decision-making process as illustrated in Figure 1 (top). This lack of faithfulness undermines the reliability of NLEs in applications where transparency and trust are paramount (Atanasova et al., 2023; Lyu et al., 2024; Parcalabescu and Frank, 2024).

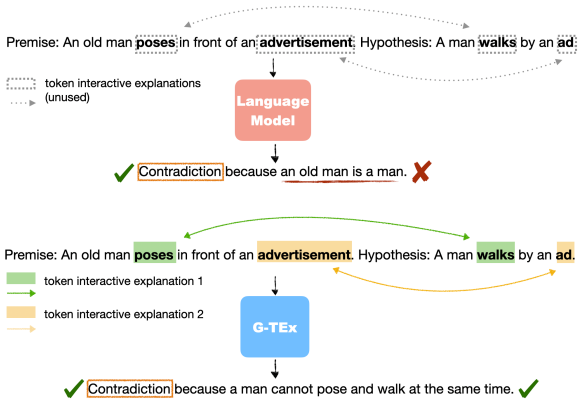

Figure 1: Faithfulness comparison between a self-rationalization model without (top) and with (bottom) the proposed G-Tex. Highlight explanations reveal the model's reasoning behind the predicted label with high faithfulness. Without G-Tex, these important tokens are omitted in the NLE while G-Tex guides the model to incorporate them in the generated NLE.

Unlike NLEs, highlight explanations reflect the model's reasoning process by identifying tokens or phrases of the input that are crucial to the model's prediction. They can be of three types: *highlight token explanations*, *token interactive explanations* and *span interactive explanations* (Sun et al., 2024) (see §3.2 for details). Though not as plausible as NLEs (Jie et al., 2024), *the faithfulness of highlight explanations is easy to measure and has been substantially improved in existing works* (Sun et al., 2024; Atanasova et al., 2020a). In this work, we hy-

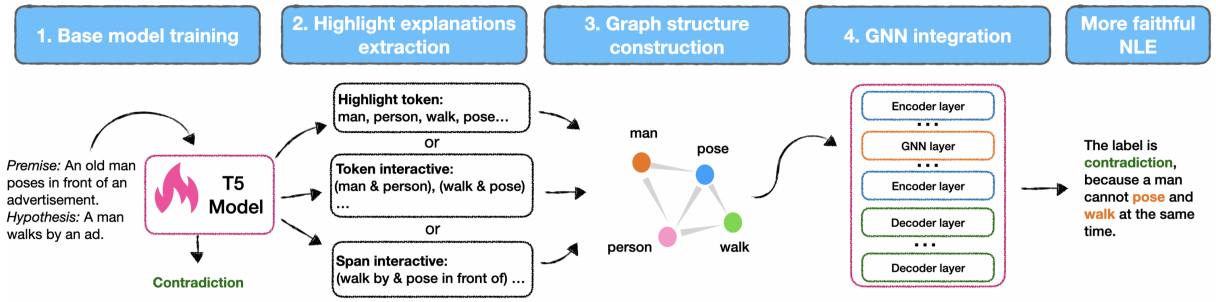

Figure 2: Illustration of our framework G-Tex, which consists of four key steps: (1) We train a base model such as T5 using the task-specific dataset for label prediction (§3.2). (2) We extract three types of highlight explanations from the trained model (§3.2). (3) We construct the graph structure based on the highlight explanations (§3.3) (4) We integrate the graph structure into the model with a GNN layer (§3.4, §3.5) and fine-tune the overall model for label prediction and NLE generation (§3.1).

pothesize that *highlight explanations can be used to improve the faithfulness of NLEs* by using them as explicit cues regarding the important parts of the input that should be present in the generated NLEs. We further hypothesize that as highlight explanations contain concise information about the most important parts of the input, they can further decrease the redundancy of NLEs and improve the overall NLE quality.

Recent efforts to improve the faithfulness of NLEs either rely on external knowledge, crafting prompts or designing the training loss for improving the faithfulness of NLEs directly (Majumder et al., 2021; Marasovic et al., 2022; Chuang et al., 2024). These methods, however, are not targeted at aligning NLEs with a model's inner reasoning but improve their faithfulness only from a model's extrinsic perspective. To address this, and inspired by Yuan et al. (2024) who leverage a Graph Neural Network (GNN) layer to guide the information flow from the input to the generation process, we propose a novel **G**raph-Guided **T**extual **Ex**planation Generation framework (G-Tex) to *enhance the faithfulness of NLEs that allows for explicitly guiding the model's reasoning with cues derived from the highly faithful highlight explanations*. The graph structure is encoded by a GNN layer, which seamlessly incorporates the highlight explanations into the NLE generation process. This also allows the model to leverage implicit anchors from the input, improving the generation of explanations.

As shown in Figure 2, we first apply a post-hoc attribution method to extract highlight explanations on a fine-tuned model based on its label prediction (§3.2). Then, we construct a graph with the most important highlight explanations for each instance (§3.3). A GNN layer is then incorporated to encode the graph within the original self-rationalization model (§3.4), which is fine-tuned to generate both the final answer prediction and the corresponding NLE simultaneously (§3.1,§3.5).

Our findings demonstrate that G-Tex substantially improves the faithfulness of NLEs by up to 12.18% compared to baselines, as evaluated on T5 (Raffel et al., 2020) and BART (Lewis et al., 2020)(see §4.2) using e-SNLI (Camburu et al., 2018), ComVE (Wang et al., 2020) and ECQA (Aggarwal et al., 2021) datasets (see §5.1). Additionally, G-Tex generates NLEs with enhanced semantic and lexical similarity, as evaluated with SacreBLEU (Post, 2018) and BERTScore (Zhang et al., 2020) respectively (see §5.2). Human evaluations further reveal improvements in decreasing redundancy and enhancing the overall quality of the generated NLEs (see details in §M). Across the different types of highlight explanations, *token and span interactive explanations* are more effective when the input text involves interaction between different parts. However, when the input consistently includes the same instruction, *highlight token explanations* prove to be more beneficial. Overall, our work introduces a novel method for explicitly guiding the NLE generation to improve faithfulness, serving as a stepping stone for addressing additional criteria for NLE and generated text.

2 Related Work

Faithfulness of Natural Language Explanations NLEs are coherent free-text explanations about the reasons behind a model's prediction. Most commonly, NLEs are produced with a self-rationalization set-up where the model generates

both a target task prediction and its NLE (Narang et al., 2020; Tang et al., 2021; Atanasova et al., 2020b; Liu et al., 2024a, 2023b,a,c, 2024b). As automatically generated NLEs suffer from faithfulness issues (Kumar and Talukdar, 2020; Wiegreffe et al., 2021; Atanasova et al., 2023; Lyu et al., 2024), existing work has explored different ways to improve that. Majumder et al. (2021) propose to first select the important parts of the input, then leverage an external commonsense knowledge generative model to get commonsense knowledge snippets about these highlights, and finally, use the soft representations of the latter for the NLE generation. Another line of work focuses on constructing suitable prompts for NLE generation (Marasovic et al., 2022). Furthermore, Wang et al. propose to prompt the model to generate the NLE and then fine-tune the LM with a counterfactual regularization loss to make the final prediction based on the generated NLE. Chuang et al. (2024) employ an estimator to provide faithfulness scores for generated NLEs. These scores and the NLEs are appended to the input and iteratively refined until the faithfulness scores converge. However, neither of these works uses direct cues from the more faithful highlight explanation for the model's prediction to guide the NLE generation, which is the novel contribution of this paper. Overall, existing work improves NLE faithfulness by resorting to external knowledge, crafting prompts or altering the generation loss. We claim that these constitute extrinsic signals, which do not directly address the NLEs' desiderata to faithfully reflect a model's inner reasoning. Our proposed method `G-Tex` directly targets this objective by guiding the generation with cues about the most important parts of the input.

Existing work has also proposed Chain-of-Thought (CoT) explanations, which reveal the model's intermediate reasoning steps before giving its final answer (Zhang et al., 2022b). These explanations can be unfaithful as well (Turpin et al., 2024; Jie et al., 2024; Lanham et al., 2023). To address this, researchers have leveraged CoT distillation techniques to train a more faithful small LM using CoT from the teacher LLM (Wang et al., 2023b; Zhang et al., 2024a; Paul et al., 2024), or have guided the original LLM to generate multiple reasoning chains and choose the most faithful one (Li et al., 2024; Jie et al., 2024). Notably, we do not focus on the CoT method for generating NLEs, as it requires specialized training data, such as reasoning chains or step-by-step intermediate explanations leading to the final answer. Moreover, CoT views faithfulness as alignment between the generated explanation and the predicted label, which differs from our focus on faithfulness to the model's internal reasoning process.

Highlight Explanations for Model Steering Prior works have found that the model's reasoning capability can be enhanced by human-annotated highlight explanations alongside the original input (Wei et al., 2022; Lampinen et al., 2022). Krishna et al. (2023) automate the process of filling the extracted highlights into few-shot templates, which enhances model accuracy across tasks such as CommonsenseQA (Talmor et al., 2019). Zhang et al. (2024b) propose iterative prompting, where the model first generates a sentence summarizing the input. This sentence is then matched with the most similar sentence from the input, with similarity calculated by an encoder, to refine the prompt and steer the model to produce an answer more accurately. Bhan et al. (2024) convert highlight explanations into NLEs using a predefined template, which is then employed to prompt the model for more accurate answers. Though they regard the NLE generation as the intermediate step, the faithfulness of these NLEs is not even evaluated. In contrast, our approach focuses on enhancing the faithfulness of the generated NLEs by integrating highlight explanations directly into the model architecture to guide NLE generation.

Graph Neural Networks for Natural Language Processing Graph neural networks (GNNs) are primarily used for graph-related tasks such as drug discovery (Han et al., 2021; Hu et al., 2021). An increasing number of researchers are exploring their potential applications in NLP tasks (Yasunaga et al., 2021; Fei et al., 2021; Lin et al., 2021). GNNs have been utilized in tasks like graph-to-text generation (Gardent et al., 2017; Yuan and Faerber, 2023) and graph-enhanced question answering (Zhang et al., 2022a), typically encoding complex graph and node representations (Koncel-Kedziorski et al., 2019). Yuan and Färber (2024) leverage GNNs to encode token-level structural information by modifying the self-attention mechanism in language models. Additionally, Yuan et al. (2024) propose a GNN-based method for information aggregation paired with a parameter-efficient fine-tuning approach. Inspired by previous work, we use GNNs to encode the highlight explanations with high faithfulness to the generation process of NLEs.

3 Methodology

In this section, we provide a detailed overview of G-Tex, as illustrated in Figure 2. We begin by introducing the self-rationalization model in §3.1. In §3.2, we describe the training of the base model for label prediction and extracting post-hoc highlight explanations as Steps 1 and 2. In Step 3 and §3.3, we outline the construction of graph structures. Finally, in Step 4, we present the GNN layer (§3.4) and explain its integration with language models (§3.5).

3.1 Overview: Self-Rationalization Model

Self-rationalization models jointly generate the task labels and NLEs to explain their reasoning for the predicted answer (Wiegreffe et al., 2021). We frame this as a text-to-text generation task. Note that we are working with tasks containing two separate parts in the input, e.g., a premise and a hypothesis on the e-SNLI dataset (see more details in §4.1). Given a sequence of tokens $x = (x_1, \ldots, x_{m+n})$ as input, where the first part of the input contains m tokens and the second part n tokens, the model M generates a label y_0 and a sequence of tokens for the NLE $y = y_0 \oplus (y_1, \ldots, y_l)$, where $\oplus$ denotes the concatenation of one label token and l NLE tokens.[1] The text generation task, encompassing both label generation and explanation generation, is implemented by a pre-trained LM with a language modeling head on top. Building on this, we insert a graph structure $\mathcal{G}$ into the standard self-rationalization model (LM) to encode the information from the highlight explanations, particularly for interactions between tokens and spans, resulting in our model M_{G-TEX} (see below). We fine-tune this model by minimizing the cross-entropy loss for the target sequence y following the same process of the standard encoder-decoder transformer model. (see Section 3.5 for details on the encoding process after integrating the GNN layer into the self-rationalization model):

$$\mathcal{L} = -\sum_{i=1}^{|y|} \log P_\phi(y_i | y_{1:i-1}, x, \mathcal{G}), \quad (1)$$

where P_ϕ is the LM's generative probability.

3.2 Post Hoc Highlight Explanation and Predicted Label

As illustrated in Figure 2, we begin by training a base model, M_{base}, designed solely to predict the label of the input text. From this model, we extract three types of highlight explanations from the input following Sun et al. (2024); Ray Choudhury et al. (2023). These highlights serve as cues revealing the model's reasoning process behind its label predictions.

Given an input instance $x = (x_1, \ldots, x_{m+n})$, each *highlight token explanation* contains one token x_i and its assigned importance score a_i; each *token interactive explanation* (x_i, x_j) consists of two interactive tokens from two separate parts of the input respectively, as well as an importance score a_{ij}; each *span interactive explanation* is formed of two spans $(span_i, span_j)$, where $span_i = (x_p, \ldots, x_{p+l_1})$ and $span_j = (x_q, \ldots, x_{q+l_2})$ are from two separate parts of the input respectively, also with an assigned importance score $a_{span_i, span_j}$, where $p, p+l_1 \in [1, m], q, q+l_2 \in [m+1, m+n]$.

Highlight Token Explanation Generation. Interactions between features in LMs are primarily captured through attention mechanisms (Vaswani, 2017). Previous work shows that highlight explanations extracted by attention-based methods show higher faithfulness than other explainability techniques (Sun et al., 2024). Building on this, we use attention weights as the basis for deriving importance scores for all types of highlight explanations. To retain the unique contributions of individual attention heads – each designed to focus on specific aspects of the data (Rogers et al., 2020) – we follow the approach of Ray Choudhury et al. (2023) to identify the most important attention head for a specific label prediction. We use the final attention layer of the model's decoder, which generates the final token representations used in generation. (see App. A for details). Subsequently, we calculate the importance score a_i for a target token x_i by averaging the self-attention scores assigned to x_i from all other tokens within the input text, following Jain and Wallace (2019); Sun et al. (2024). The extracted *highlight token explanation* set for instance x is noted as $HT = \{(x_i, a_i) | i \in [1, m+n]\}$.

Token Interactive Explanation Generation. Using the most important attention head identified as described above, we calculate the importance score a_{ij} for each *token interactive explanation* by averaging the attention weights between these two tokens x_i and x_j following Clark et al. (2019). The *token interactive explanation* set for instance x is $TI = \{((x_i, x_j), a_{ij}) | i \in [1, m], j \in [m+1, n])\}$.

[1] See App. C input and output example for e-SNLI.

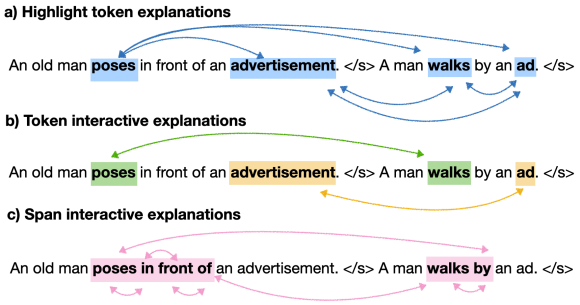

Figure 3: We generate three different types of post-hoc highlight explanations and use them to construct graph structures guiding the NLE generation within our framework. For simplicity, we present only a subset of the explanations for each type.

Span Interactive Explanation Generation. Since *token interactive explanations* may not convey meaningful information on their own, Ray Choudhury et al. (2023) suggest using span interactions, which consist of more coherent phrases and are found to be more plausible (Sun et al., 2024). Following their approach, we apply the Louvain algorithm (Blondel et al., 2008) to extract *span interactive explanations* by identifying communities of token interactions. Tokens are treated as nodes, with the importance scores of token pair interactions used as edge weights. The communities of token interactions are selected to have dense intra-span and sparse inter-span interactions. For each x, span pairs $(span_i, span_j)$ are extracted, and the importance score $a_{span_i,span_j}$ for each span pair is computed by averaging the importance scores of the constituent token pairs. The set of generated *span interactive explanations* is denoted as $SI = \{(x_{span_i,span_j}, a_{span_i,span_j}) | span_i = (x_p, \ldots, x_{p+l_1}), span_j = (x_q, \ldots, x_{q+l_2})\}$. The number of generated span pairs depends on the community detection algorithm and is $< m! * n!$ since only neighboring tokens within the same community can form spans, and spans must come from different parts of the input to form valid pairs.

3.3 Post Hoc Highlight Explanations as a Graph

We build graph structures based on the three different types of highlight explanations (see Figure 3). Notably, we treat each token as a node in the graph structure and assign edges between the extracted tokens. Following Yuan and Färber (2024), an edge is also assigned to connect the subtokens if a word is tokenized into several subtokens.

Highlight Token Explanation We use the importance scores derived in Section §3.2 to select the top-k% most important highlight token explanations, as less important tokens might introduce noise. Then we assign equally weighted bidirectional edges between these tokens to ensure information flow among them (see Figure 3a).

Token Interactive Explanations We also select the top-k% token interactive explanations with the highest importance scores. Then equally weighted bidirectional edges are assigned to connect the tokens within each token interaction (see Figure 3b).

Span Interactive Explanation As only a few spans are extracted from the input text as described in Section 3.2, all the interactive spans are used to construct the graph structure. Within a span, all subtokens are connected. Between spans, tokens are connected with each other (see Figure 3c).

3.4 Graph Neural Network Layer

The GNN layer aggregates information of highlight explanations to model graph and node representations based on the graph structures as introduced in §3.3. We define a bidirectional graph $\mathcal{G}$ as a triple $(\mathcal{V}, \mathcal{E}, \mathcal{R})$ with a set of nodes $\mathcal{V} = \{v_1, \ldots, v_n\}$ (one node for each token), a set of relation types $\mathcal{R}$[2]., and a set of edges $\mathcal{E}$ of the form (v, r, v') with $v, v' \in \mathcal{V}$, and $r \in \mathcal{R}$. Each node v_i is associated with a feature vector h_i, which represents the hidden states of the i-th token in the l-th layer.

The node representations in the GNN layer are updated by aggregating information from neighboring nodes by different aggregation algorithms depending on the chosen GNN architecture. In our work, we employ three most representative and widely used GNN architectures following previous work (Yuan et al., 2024; Yuan and Färber, 2024): Graph Convolutional Network (GCN, Kipf and Welling (2017)), Graph Attention Network (GAT, Veličković et al. (2018)) and GraphSAGE (Hamilton et al., 2017). While GCN aggregates information from neighboring nodes uniformly, GAT introduces attention weights to prioritize and aggregate incoming information.[3] GraphSAGE, on the other hand, incorporates information from the

[2]We consider only one type of relation: the bidirectional edge between nodes v and v', with all edges weighted equally for initialization, note that the edge values will update during fine-tuning

[3]Details of the learning processes for GCN and GAT are provided in App. D.

current node and its neighboring nodes as follows:

$$h_v = \sigma\left(W\left(h_v^{(l)} \oplus \text{AGG}(\{h_{v'}^{(l)}, \forall v' \in N(v)\})\right)\right) \quad (2)$$

where h_v denotes the updated node representation of v, $h_{v'}^{(l)}$ is the token representation of its neighbouring nodes from l-th layer, σ the activation function, W are the trainable parameters of the GNN, $N(v)$ includes all the neighbouring nodes of v. The concatenation function $\oplus$ concatenates aggregated information with the node's current representation, and the aggregation function AGG aggregates the information flowing from the neighboring nodes using techniques such as mean, pool, and LSTM.[4]

3.5 Integrating GNN in Language Models

As illustrated in Figure 2, Step 4, we integrate a GNN layer into the LM by stacking it on top of the n-th encoder layer. Yuan et al. (2024) demonstrated that incorporating a GNN into LLMs is most effective when placed in the last three-quarters of the layers, following the principles of information flow theory (Wang et al., 2023a). In line with prior work, we similarly position the GNN layer at the $3/4$-th encoder layer. The GNN layer takes token representations from the l-th encoder layer, processes them along with graph structures derived from highlight explanations, and then forwards the augmented representations h_v to the next encoder layer $l+1$, which can be formulated as:

$$\tilde{h}^{(l)} = \text{LayerNorm}(h_v + \text{Attention}(h_v W^Q, h_v W^K, h_v W^V)) \quad (3)$$

$$h^{(l+1)} = \text{LayerNorm}(\tilde{h}^{(l)} + \text{FFN}(\tilde{h}^{(l)})) \quad (4)$$

The rest of the model architecture remains unchanged.

4 Experiments

4.1 Datasets

We use three widely adopted reasoning datasets with human-annotated explanations: **e-SNLI** (Camburu et al., 2018), **ComVE** (Wang et al., 2020) and **ECQA** (Aggarwal et al., 2021). **e-SNLI** extends SNLI with human-annotated explanations for each premise-hypothesis pair, providing both the correct label (entailment, contradiction, or neutral) and a human-annotated NLE for why the label was chosen. **ComVE** provides natural language explanations identifying which of the two provided statements contradicts common sense. **ECQA** is a multiple-choice question-answering dataset with human-annotated explanations for each choice.[5]

4.2 Experimental Setting

We select two commonly used models for self-rationalization (Raffel et al., 2020; Narang et al., 2020; Marasovic et al., 2022; Lewis et al., 2020; Huang et al., 2023; Yadav et al., 2024), `T5-large` and `BART-large` as our base models, both of which follow an encoder-decoder architecture. For these models, we insert the graph at the $3/4$-th encoder layer. We are not targeting the decoder-only models as they rely solely on the previous token rather than graph embeddings of all tokens for next-token prediction, which limits the guidance of the highlight explanation graphs, and we encourage the modification to apply to decoder-only models for future work (See Limitations). Our `G-Tex` is fine-tuned on the training set, with validation performed on the validation set at each epoch. The BLEU score (Papineni et al., 2002) is used to select the best-performing checkpoint. Further experimental details can be found in App. F.

4.3 Models

We use two baselines in our experiments to compare against `G-Tex`:

`Fine-tuning`$_{\text{base}}$ We fine-tune the base models `T5-large` and `BART-large` on the training set of e-SNLI and ECQA for self-rationalization.

`Prompt` To incorporate highlight explanations as part of the input, we concatenate the template, *"The most important tokens are: token$_1$, token$_2$, token$_3$, ..."* to the end of the input sentence and fine-tune the models accordingly. The important tokens are extracted from the highlight explanations, consistent with the top-k% tokens used in `G-Tex`.

`G-Tex` For our approach, we utilize the encoder-decoder model `T5-large` and `BART-large` as the base models and insert a GNN layer after the $3/4$-th encoder layer. This GNN layer injects the structured information from the highlight explanations. We experiment with three distinct types of GNN architectures, which we denote as **`Tex-GCN`**, **`Tex-GAT`**, and **`Tex-SAGE`**, representing Graph Convolutional

[4] Mean aggregation is applied to GraphSAGE in this work.

[5] In order to explore how different highlight explanations affect faithfulness, we reformulate e-SNLI, ECQA and ComVE into different formats. While the input for e-SNLI and ECQA consists of two distinct sentences, ComVE always includes the same question as the first part of the input (see examples in App. C). This distinction is to explore whether the interaction between the two input parts is significant.

Explanation Type	Model	e-SNLI				ComVE			
		Unfaithfulness(%↓)		Automatic(↑)		Unfaithfulness(%↓)		Automatic(↑)	
		Counter	Total	SacreBLEU	BERTScore	Counter	Total	SacreBLEU	BERTScore
T5-based									
-	**Fine-tuning$_{base}$**	47.70 ±2.31	17.68 ±1.94	15.430	0.894	92.37 ±1.21	68.96 ±2.23	7.634	0.876
Highlight Token	**Prompt**	43.61 ±2.86	14.71 ±1.16	15.686	0.898	93.25 ±1.19	68.90 ±2.61	7.592	0.876
	Tex-SAGE (Ours)	33.83 ±1.51	11.07 ±1.14	16.426	**0.908**	90.53 ±1.40	**57.48** ±0.58	**9.016**	0.884
Token Interactions	**Prompt**	54.36 ±3.11	20.60 ±1.81	15.478	0.898	**87.39** ±1.78	77.71 ± 2.06	7.028	0.888
	Tex-SAGE (Ours)	34.27 ±1.63	11.00 ±1.66	**16.443**	**0.908**	87.47 ±2.21	76.94 ± 2.33	6.956	0.888
Span Interactions	**Prompt**	42.86 ±2.20	13.19 ±1.95	16.031	0.899	89.90 ±0.86	79.70 ±2.15	7.226	0.889
	Tex-SAGE (Ours)	**33.25** ±2.18	**10.08** ±2.02	16.277	0.907	89.64 ±0.91	76.39 ±3.36	7.652	**0.891**
BART-based									
-	**Fine-tuning$_{base}$**	57.71 ±2.39	22.52 ±1.86	15.732	0.906	91.09 ±1.81	70.50 ±1.68	10.070	**0.891**
Highlight Token	**Prompt**	57.52 ±3.84	24.45 ±0.62	15.678	0.898	90.23 ±2.10	68.82 ±2.97	10.012	0.876
	Tex-SAGE (Ours)	**44.72** ±4.71	14.75 ±2.13	16.318	**0.909**	**87.91** ±2.74	**58.32** ±0.81	**10.552**	0.884
Token Interactions	**Prompt**	47.73 ±3.16	19.59 ±1.72	15.478	0.898	89.80 ±4.54	69.43 ±3.14	7.215	0.888
	Tex-SAGE (Ours)	46.88 ±3.34	15.68 ±1.75	**16.427**	**0.909**	88.15 ±2.47	68.08 ± 2.47	7.333	0.888
Span Interactions	**Prompt**	50.98 ±3.72	18.34 ±1.70	16.027	**0.909**	95.17 ±1.18	64.35 ± 0.94	7.953	0.889
	Tex-SAGE (Ours)	45.17 ±3.52	**14.64** ±1.32	16.517	**0.909**	94.29 ±2.57	63.76 ± 2.49	7.953	**0.891**

Table 1: Overall evaluation results on e-SNLI and ComVE datasets for T5-based and BART-based models, with our **G-Tex** model using **Tex-SAGE**. Counter indicates *Counter Unfaith*, Total indicates *Total Unfaith*, with both the mean values and standard deviations reported from 5 runs with different random seeds. The p-values (Wasserstein and Lazar, 2016) can be found in Appendix §J, Table 6. The best performance of each evaluation metric is in bold. See Appendix §K for results on ECQA dataset and Appendix §I, Table 5 for results of our model using **Tex-GAT** and **Tex-GCN**.

Networks, Graph Attention Networks, and GraphSAGE, respectively (see §3.4).

5 Evaluation

We conduct a comprehensive evaluation of the models, using a faithfulness test, automatic metrics and human assessment on multiple dimensions[6]. As for the label predictions, G-Tex achieves results that are better or comparable to the baselines. We report an overview of the label prediction performance in Table 3, App. E.

5.1 Faithfulness Evaluation

To assess the faithfulness of the generated NLEs, we apply the counterfactual faithfulness test from Atanasova et al. (2023). This method involves inserting random adjectives in front of nouns of the original input, resulting in multiple perturbed instances. If the model's prediction changes, the newly generated NLE should include the inserted word; otherwise, the original NLE is unfaithful as it is potentially misaligned with the model's reasoning. Note that the unchanged label provides no relevant information about the faithfulness of the NLE. See details in App. H.

Following Atanasova et al. (2023), we apply this test on the e-SNLI, ComVE and ECQA datasets, calculating: (1) the percentage of instances where, for at least one altered input, the inserted word does not appear in the new NLE across instances with label change(*Counter Unfaith*); and (2) the proportion of these unfaithful instances across all instances (*Total Unfaith*).

Results As shown in Tables 1, our G-Tex[7] We present results on e-SNLI and ComVE as representative datasets for NLI and commonsense QA, respectively, and defer ECQA results to App. K. with T5 as the base model leads up to 9.60% decrease in *Total Unfaithful* on e-SNLI (20.60% vs. 11.00% with token interactive explanations) and up to 11.48% on ComVE (68.96% vs. 57.48% with highlight tokens) compared to the Fine-tuning$_{base}$ and Prompt. Similarly, G-Tex with BART as the base model leads up to a 9.70% decrease in *Total Unfaithful* on e-SNLI (24.45% vs. 14.75% with highlight explanations) and up to 12.18% decrease on ComVE (70.50% vs. 58.32% with highlight explanations). While G-Tex with T5 slightly under-

[6]The results and analysis of human evaluation are presented in App. M

[7]We select **Tex-SAGE** to present the results for G-Tex, as GraphSAGE demonstrates superior performance in modeling text-based graph structures according to previous work (Yuan and Färber, 2024). The results of other G-Tex models and the discussion across all GNN variants can be found in App. I.

performs the prompt baseline on ComVE with *token interactive explanations*, overall, **our method outperforms all baselines in counterfactual unfaithfulness and total faithfulness**.

Across the different highlight explanation types, different datasets yield different results. On the e-SNLI dataset, *span interactive explanations* produce more faithful NLEs with T5-based models (10.08% *Total Unfaith*). For the e-SNLI task, the input text consists of two parts, namely the premise and the hypothesis, and interactive explanations between these parts are of paramount importance in indicating the reasoning process of the models. **Thus, *token interactive* and *span interactive explanations* tend to improve faithfulness more effectively than *highlight token explanations***. This aligns with previous work showing that these highlight explanations offer higher faithfulness in recovering a model's prediction (Sun et al., 2024).

However, *highlight token explanations* also show significant benefits when the task input consists of the same instruction/first part. As the first part of the input for ComVE is formulated as the same question, the second part of the input becomes especially important in distinguishing the input text for the models. The results on ComVE indicate that *highlight token explanations* yield the lowest *Total Unfaith* for both T5- and BART-based `G-Tex` (57.48% and 58.32%, respectively). **Thus, *highlight token explanations* can improve the faithfulness when the interaction between parts of the input is less critical.**

Our findings demonstrate that while all highlight explanations are significantly important, their utility depends on the task. When the input text involves interaction between different parts, *token and span interactive explanations* are more useful. However, when the input consistently includes the same instruction, *highlight token explanations* are more effective. Nonetheless, regardless of the task, the results again verify that `G-Tex` effectively leverages different types of highlight explanations for NLE generation, leading to more faithful NLEs.

5.2 Automatic Metrics for Similarity between NLEs and Golden explanations

To assess the alignemnt of generated NLEs with human-written ones, we measure the similarity between them and the golden human-annotated explanations. A similarity with human-written explanations is used in existing work to indicate how plausible the generated NLEs would appear to end users (Sun et al., 2024). We employ automatic evaluation metrics **SacreBLEU** (Post, 2018) and **BERTScore** (Zhang et al., 2020) to capture both lexical and semantic similarity.[8]

As shown in Table 1, the automatic evaluation results demonstrate that `G-Tex` generates NLEs of higher alignment with human-written explanations in terms of lexical and semantic similarity on the e-SNLI dataset, outperforming the `Fine-tuning`$_{\text{base}}$ and `Prompt`. Across all explanation types, `G-Tex` consistently achieves higher SacreBLEU scores, such as 16.443 for `G-Tex` with the *token interactive explanation* setting, and better BERTScores, such as 0.909 across most BART-based methods. Regarding the ComVE dataset, `G-Tex` also generates NLEs with higher SacreBLEU and BERTScore. For BART-based `G-Tex`, the highest ScareBLEU is 10.552 achieved with `G-Tex` with *highlight token explanations*. **These results demonstrate that our models generate explanations with improved alignment with human explanations**. Furthermore, they confirm that interactive explanations are more effective for e-SNLI, while highlight token explanations are more beneficial for ComVE, due to the distinct structure of their inputs.

6 Conclusion

In this work, we propose `G-Tex`, a novel framework that incorporates the reasoning process of models to enhance faithfulness in NLEs. `G-Tex` allows for integrating various types of highlight explanations through a GNN layer within language models. Evaluated via faithfulness tests, automatic metrics, and human evaluation on three reasoning datasets, `G-Tex` demonstrates consistent improvements in faithfulness, alignment with human-annotated explanations, and reduced redundancy. Our results show that the benefits of different highlight explanations depend on task formulation: *token and span interactive explanations* work best for tasks requiring input interaction, while *highlight token explanations* are more effective when interactions are less critical. These findings highlight the potential of `G-Tex` as an interpretable framework that embeds the reasoning process of language models as a graph structure to improve model faithfulness. Future work could explore various graph structures and explanation types to further enhance the versatility and effectiveness of `G-Tex` for larger models.

[8] In addition to SacreBLEU and BERTScore, results for other automatic metrics are provided in App. L.

Limitations

Our work proposes a novel graph-guided framework for natural language explanation generation, utilizing highlight explanations in the form of highlight tokens, token interactives, and span interactives. While G-Tex improves the models' faithfulness constantly, we acknowledge several limitations in our approach.

Firstly, we applied G-Tex exclusively to encoder-decoder models. This choice was made not only because encoder-decoder models are better suited for text-to-text format tasks, but also because the encoder is able to embed the graph structure and utilize it to generate each individual token. While our approach is potentially applicable to decoder-only models, their architectural differences introduce notable complexities. In decoder-only models, token generation relies solely on the hidden states of the preceding token. As a result, significant adjustments, such as carefully integrating text embeddings with graph embeddings, would be required to adapt our method for use with decoder-only models to generate tokens, which is thus outside the scope of this work. Due to limited computational resources, we chose T5-large and BART-large as the models to fine-tune for NLE generation. Their established reasoning capabilities and relatively lightweight nature make them well-suited for our experimental setup. We encourage future work to explore how model scalability affects the quality of generated NLEs.

Secondly, while G-Tex leverages the reasoning process of the models and offers a more transparent and interpretable framework, the internal mechanisms of the GNN layer remain unexplored in this study. Moreover, we use specific graph types to construct the highlight explanations, assigning equal weights to the edges between nodes. Future work could explore weighted edges and alternative graph structures to encode highlight explanations.

Thirdly, while we choose the attention-based methods as the foundation to extract highlight explanations due to their higher faithfulness on ECQA and e-SNLI dataset Sun et al. (2024), it is important to acknowledge other important explainability techniques, such as perturbation-based attribution e.g., Shapley (Lundberg and Lee, 2017)), Integrated Gradients (Sundararajan et al., 2017; Serrano and Smith, 2019) and Saliency Map (Feldhus et al., 2022). It is worth exploring how the highlight explanations generated by different explainability techniques impact the quality of generated NLEs on broader datasets. We leave this exploration for future work.

Lastly, we evaluate the quality of NLEs generated by our model using three reasoning datasets, e-SNLI (NLI task), ComVE and ECQA (commonsense QA task). As more datasets meeting these criteria become accessible in the future, we encourage further exploration of our method in additional domains.

References

Shourya Aggarwal, Divyanshu Mandowara, Vishwajeet Agrawal, Dinesh Khandelwal, Parag Singla, and Dinesh Garg. 2021. Explanations for CommonsenseQA: New Dataset and Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3050–3065, Online. Association for Computational Linguistics.

David Alvarez Melis and Tommi Jaakkola. 2018. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Pepa Atanasova, Oana-Maria Camburu, Christina Lioma, Thomas Lukasiewicz, Jakob Grue Simonsen, and Isabelle Augenstein. 2023. Faithfulness tests for natural language explanations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 283–294, Toronto, Canada. Association for Computational Linguistics.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020a. A diagnostic study of explainability techniques for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, Online. Association for Computational Linguistics.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020b. Generating fact checking explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, Online. Association for Computational Linguistics.

Milan Bhan, Jean-Noël Vittaut, Nicolas Chesneau, and Marie-Jeanne Lesot. 2024. Self-AMPLIFY: Improving small language models with self post hoc explanations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10974–10991, Miami, Florida, USA. Association for Computational Linguistics.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast Unfolding of Communities in Large Networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 552–562, Hong Kong, China. Association for Computational Linguistics.

Yu-Neng Chuang, Guanchu Wang, Chia-Yuan Chang, Ruixiang Tang, Shaochen Zhong, Fan Yang, Mengnan Du, Xuanting Cai, and Xia Hu. 2024. FaithLM: Towards Faithful Explanations for Large Language Models. *Preprint*, arXiv:2402.04678.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Zichu Fei, Qi Zhang, and Yaqian Zhou. 2021. Iterative GNN-based decoder for question generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2573–2582, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nils Feldhus, Leonhard Hennig, Maximilian Dustin Nasert, Christopher Ebert, Robert Schwarzenberg, and Sebastian Möller. 2022. Constructing natural language explanations via saliency map verbalization. *arXiv preprint arXiv:2210.07222*.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Kehang Han, Balaji Lakshminarayanan, and Jeremiah Zhe Liu. 2021. Reliable graph neural networks for drug discovery under distributional shift. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2021. Open graph benchmark: Datasets for machine learning on graphs. *Preprint*, arXiv:2005.00687.

Fan Huang, Haewoon Kwak, and Jisun An. 2023. Chain of Explanation: New Prompting Method to Generate Quality Natural Language Explanation for Implicit Hate Speech. In *Companion Proceedings of the ACM Web Conference 2023*, pages 90–93.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Yeo Wei Jie, Ranjan Satapathy, Rick Goh, and Erik Cambria. 2024. How Interpretable are Reasoning Explanations from Prompting Large Language Models? In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2148–2164.

Shailza Jolly, Pepa Atanasova, and Isabelle Augenstein. 2022. Generating Fluent Fact Checking Explanations with Unsupervised Post-editing. *Information*, 13(10):500.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2019. Text Generation from Knowledge Graphs with Graph Transformers. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2284–2293, Minneapolis, Minnesota. Association for Computational Linguistics.

Satyapriya Krishna, Jiaqi Ma, Dylan Z Slack, Asma Ghandeharioun, Sameer Singh, and Himabindu Lakkaraju. 2023. Post hoc explanations of language models can improve language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Sawan Kumar and Partha Talukdar. 2020. NILE: Natural Language Inference with Faithful Natural Language Explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8730–8742.

Andrew Lampinen, Ishita Dasgupta, Stephanie Chan, Kory Mathewson, Mh Tessler, Antonia Creswell, James McClelland, Jane Wang, and Felix Hill. 2022. Can Language Models Learn from Explanations in

Context? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 537–563.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. 2023. Measuring Faithfulness in Chain-of-Thought Reasoning. *arXiv preprint arXiv:2307.13702*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jiachun Li, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2024. Towards Faithful Chain-of-Thought: Large Language Models are Bridging Reasoners. *arXiv preprint arXiv:2405.18915*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yuxiao Lin, Yuxian Meng, Xiaofei Sun, Qinghong Han, Kun Kuang, Jiwei Li, and Fei Wu. 2021. BertGCN: Transductive text classification by combining GNN and BERT. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1456–1462, Online. Association for Computational Linguistics.

Wei Liu, Zhiying Deng, Zhongyu Niu, Jun Wang, Haozhao Wang, YuanKai Zhang, and Ruixuan Li. 2024a. Is the MMI criterion necessary for interpretability? degenerating non-causal features to plain noise for self-rationalization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Wei Liu, Haozhao Wang, Jun Wang, Zhiying Deng, Yuankai Zhang, Cheng Wang, and Ruixuan Li. 2024b. Enhancing the rationale-input alignment for self-explaining rationalization. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 2218–2230. IEEE.

Wei Liu, Haozhao Wang, Jun Wang, Ruixuan Li, Xinyang Li, YuanKai Zhang, and Yang Qiu. 2023a. MGR: Multi-generator based rationalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12771–12787, Toronto, Canada. Association for Computational Linguistics.

Wei Liu, Jun Wang, Haozhao Wang, Ruixuan Li, Zhiying Deng, YuanKai Zhang, and Yang Qiu. 2023b. D-separation for causal self-explanation. In *Advances in Neural Information Processing Systems*, volume 36, pages 43620–43633. Curran Associates, Inc.

Wei Liu, Jun Wang, Haozhao Wang, Ruixuan Li, Yang Qiu, Yuankai Zhang, Jie Han, and Yixiong Zou. 2023c. Decoupled rationalization with asymmetric learning rates: A flexible lipschitz restraint. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1535–1547.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Scott M Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. *Advances in neural information processing systems*, 30.

Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2024. Towards Faithful Model Explanation in Nlp: A Survey. *Computational Linguistics*, pages 1–67.

Bodhisattwa Prasad Majumder, Oana-Maria Camburu, Thomas Lukasiewicz, and Julian McAuley. 2021. Knowledge-Grounded Self-Rationalization via Extractive and Natural Language Explanations. *arXiv preprint arXiv:2106.13876*.

Ana Marasovic, Iz Beltagy, Doug Downey, and Matthew Peters. 2022. Few-shot self-rationalization with natural language prompts. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 410–424, Seattle, United States. Association for Computational Linguistics.

Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. Wt5?! Training Text-to-Text Models to Explain Their Predictions. *arXiv preprint arXiv:2004.14546*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Letitia Parcalabescu and Anette Frank. 2024. On measuring faithfulness or self-consistency of natural language explanations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6048–6089, Bangkok, Thailand. Association for Computational Linguistics.

Debjit Paul, Robert West, Antoine Bosselut, and Boi Faltings. 2024. Making Reasoning Matter: Measuring and Improving Faithfulness of Chain-of-Thought Reasoning. *arXiv preprint arXiv:2402.13950*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with A Unified Text-to-Text Transformer. *Journal of machine learning research*, 21(140):1–67.

Sagnik Ray Choudhury, Pepa Atanasova, and Isabelle Augenstein. 2023. Explaining interactions between text spans. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12709–12730, Singapore. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Martin Schmitt, Hinrich Schütze, and Iryna Gurevych. 2021. Investigating pretrained language models for graph-to-text generation. In *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI*, pages 211–227, Online. Association for Computational Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy. Association for Computational Linguistics.

Jingyi Sun, Pepa Atanasova, and Isabelle Augenstein. 2024. A unified framework for input feature attribution analysis. *Preprint*, arXiv:2406.15085.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic Attribution for Deep Networks. In *International conference on machine learning*, pages 3319–3328. PMLR.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Xuejiao Tang, Xin Huang, Wenbin Zhang, Travers B Child, Qiong Hu, Zhen Liu, and Ji Zhang. 2021. Cognitive Visual Commonsense Reasoning Using Dynamic Working Memory. In *Big Data Analytics and Knowledge Discovery: 23rd International Conference, DaWaK 2021, Virtual Event, September 27–30, 2021, Proceedings 23*, pages 81–93. Springer.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2024. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. *Advances in Neural Information Processing Systems*, 36.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

Cunxiang Wang, Shuailong Liang, Yili Jin, Yilong Wang, Xiaodan Zhu, and Yue Zhang. 2020. SemEval-2020 task 4: Commonsense validation and explanation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 307–321, Barcelona (online). International Committee for Computational Linguistics.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023a. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9840–9855, Singapore. Association for Computational Linguistics.

PeiFeng Wang, Aaron Chan, Filip Ilievski, Muhao Chen, and Xiang Ren. PINTO: Faithful Language Reasoning Using Prompt-Generated Rationales. In *Workshop on Trustworthy and Socially Responsible Machine Learning, NeurIPS 2022*.

Peifeng Wang, Zhengyang Wang, Zheng Li, Yifan Gao, Bing Yin, and Xiang Ren. 2023b. SCOTT: Self-Consistent Chain-of-Thought Distillation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5546–5558.

Ronald L Wasserstein and Nicole A Lazar. 2016. The asa statement on p-values: context, process, and purpose.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Advances in neural information processing systems*, 35:24824–24837.

Sarah Wiegreffe, Ana Marasović, and Noah A Smith. 2021. Measuring Association Between Labels and Free-Text Rationales. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10266–10284.

Neemesh Yadav, Sarah Masud, Vikram Goyal, Md Shad Akhtar, and Tanmoy Chakraborty. 2024. Tox-BART: Leveraging Toxicity Attributes for Explanation Generation of Implicit Hate Speech. *arXiv preprint arXiv:2406.03953*.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. QA-GNN: Reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter*

of the Association for Computational Linguistics: Human Language Technologies, pages 535–546, Online. Association for Computational Linguistics.

Shuzhou Yuan and Michael Faerber. 2023. Evaluating generative models for graph-to-text generation. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 1256–1264, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Shuzhou Yuan and Michael Färber. 2024. GraSAME: Injecting token-level structural information to pretrained language models via graph-guided self-attention mechanism. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 920–933, Mexico City, Mexico. Association for Computational Linguistics.

Shuzhou Yuan, Ercong Nie, Michael Färber, Helmut Schmid, and Hinrich Schuetze. 2024. GNNavi: Navigating the information flow in large language models by graph neural network. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 3987–4001, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems*, volume 34, pages 27263–27277. Curran Associates, Inc.

Jiang Zhang, Qiong Wu, Yiming Xu, Cheng Cao, Zheng Du, and Konstantinos Psounis. 2024a. Efficient Toxic Content Detection by Bootstrapping and Distilling Large Language Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21779–21787.

Qingru Zhang, Xiaodong Yu, Chandan Singh, Xiaodong Liu, Liyuan Liu, Jianfeng Gao, Tuo Zhao, Dan Roth, and Hao Cheng. 2024b. Model Tells Itself Where to Attend: Faithfulness Meets Automatic Attention Steering. *arXiv preprint arXiv:2409.10790*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

X Zhang, A Bosselut, M Yasunaga, H Ren, P Liang, C Manning, and J Leskovec. 2022a. GreaseLM: Graph REASoning Enhanced Language Models for Question Answering. In *International Conference on Representation Learning (ICLR)*.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022b. Automatic Chain of Thought Prompting in Large Language Models. *arXiv preprint arXiv:2210.03493*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 563–578, Hong Kong, China. Association for Computational Linguistics.

A Post Hoc Explanation Generation Details

For each attention head j regarding generating token k, When the contribution of input token i c_{ji} is positive, the larger the weight w_{ji}, the more important of input token i to k. We aggregate the importances for generating k from all input tokens in attention head j as the indication of the overall importance of attention head j.

B Raw Running Time of Exacting Highlight Explanations

We report the raw running time for extracting highlight explanations on the test set of e-SNLI using T5-based model in Table 2. Although the span interactive explanation has the longest runtime, it only requires 14 ms to extract explanations for an instance with the longest token range. While extracting explanations adds some computational time, it is not prohibitive for practical use.

C Example of Self-Rationalization Task

One example of the e-SNLI dataset (Camburu et al., 2018) will be:

> ***Input***: *"Premise: A woman is asleep at home. Hypothesis: A woman with a red scarf is giving a shushing sign to the camera in front of shelves of books."*
> ***Output***: *"Contradiction. The woman cannot be giving a sign and asleep at the same time."*

where *Contradiction* is the predicted label, and the text explains the reason for the prediction.

And we reformulate ComVE dataset (Wang et al., 2020) as:

> ***Input***: *"Which statement of the two is against common sense?*
> *1. when it is hot humidity forms*
> *2. when it rains humidity forms*
> *amaranthine"*
> ***Output***: *"2. Water makes humidity, not temperature."*

Explanation Type	[5, 20) Tokens	[20, 40) Tokens	[40, 69] Tokens
Average Time Cost per Instance (ms)			
Highlight Token Explanation	0.7382	1.2903	2.5249
Token Interactive Explanation	0.3924	0.8071	1.8725
Span Interactive Explanation	2.5501	5.8975	14.3293
Number of Instances in Each Token Range			
	4,546	5,068	192

Table 2: Average time cost (in milliseconds) and instance counts across different token length ranges for three types of highlight explanation extraction using a T5-based model on the e-SNLI test set.

where *2* is the index of the prediction, and the text explains the reason why it is against common sense.

Meanwhile, one example of ECQA (Aggarwal et al., 2021):

> ***Input***: *"The student was contemplating the problem, that's when he made the what that led him to the answer? action, discovery, reflection, deciding, thinking*
> ***Output***: *"discovery. Contemplating on the problem, the student made the discovery or finding that led him to the answer. Contemplating and thinking deeply about the problem may or may not lead to action. Reflection is contemplating of thinking about oneself and not the problem. Deciding is contemplating choice and wrong decisions don't lead to answer. Thinking won't necessarily lead to the answer."*

where *discovery* is the predicted answer, and the text explains the reason why it is correct and why the others are wrong.

D Aggregation Algorithms of GCN and GAT

The learning process of GCN is formulated as:

$$h_v = \sigma\left(W \sum_{v' \in N(v)} \frac{h_{v'}^{(l)}}{|N(v)|}\right) \tag{5}$$

where h_v denotes the updated node representation of v, $h_{v'}^{(l)}$ is the token representation of its neighbouring nodes from l-th layer, σ the activation function, W are the trainable parameters of the GNN, $N(v)$ includes all the neighbouring nodes of v.

Unlike the average over all neighbouring nodes in GCN, GAT learns an attention weight α for every neighbouring node:

$$h_v = \sigma\left(\sum_{v' \in N(v)} \alpha_{vv'} W h_{v'}^{(l)}\right) \tag{6}$$

E Performance for Label Prediction

We present the performance of all baselines and `G-Tex` for the label prediction task in Table 3. `G-Tex` consistently outperforms the baselines on both the e-SNLI and ECQA datasets.

As shown in Table 3, we present our `G-Tex` models' performance in answer prediction, where the GNN layer is jointly fine-tuned with the base model alongside all baseline models. It is evident that the `G-Tex` model achieves better or comparable accuracy to the baseline models, ensuring that `G-Tex` does not sacrifice answer accuracy while increasing NLE faithfulness.

F Experimental Details

The number of incorporated GNN layers is 1. Final results are reported on the test set with beam search set to 3. We set $k = 30$ to take the top 30% most important highlight explanations. Training is conducted on four NVIDIA A100-SXM4-40GB GPUs, utilizing AdamW (Loshchilov and Hutter, 2019) as the optimizer. The learning rate is set to `3e-4` for both the baselines and `G-Tex` after grid search. And beam search is set to 3 for the text generation. We use the original train, dev, and test splits for model fine-tuning across all the datasets.

G Model Size

Table 4 shows the number of trainable parameters comprising the baselines and G-Tex, as well as the training time for one epoch under the same configuration (batch size, optimizer, learning rate, etc.).

Method		$Acc_{e\text{-}SNLI}$	Acc_{ECQA}	Acc_{ComVE}
T5-large				
Fine-tuning$_{base}$		84.50	61.56	89.92
Highlight Tokens	Prompt	86.16	60.98	88.05
	Tex-GCN	89.79	59.87	90.86
	Tex-GAT	89.42	60.22	91.08
	Tex-SAGE	89.78	60.37	92.43
Token Interactions	Prompt	86.02	57.17	90.48
	Tex-GCN	89.88	**62.23**	90.97
	Tex-GAT	89.93	61.76	90.14
	Tex-SAGE	89.94	61.25	89.76
Span Interactions	Prompt	88.92	59.14	88.14
	Tex-GCN	89.76	59.62	89.06
	Tex-GAT	89.10	59.02	90.36
	Tex-SAGE	89.98	58.62	89.76
BART-large				
Fine-tuning$_{base}$		85.29	56.91	91.57
Highlight Tokens	Prompt	81.55	42.21	91.47
	Tex-GCN	**91.04**	41.82	92.17
	Tex-GAT	90.60	50.50	92.15
	Tex-SAGE	91.03	52.73	**92.67**
Token Interactions	Prompt	90.42	54.59	90.48
	Tex-GCN	90.18	58.02	91.76
	Tex-GAT	89.52	55.50	86.51
	Tex-SAGE	89.44	52.46	91.77
Span Interactions	Prompt	90.35	56.38	89.13
	Tex-GCN	90.91	51.53	91.77
	Tex-GAT	91.03	56.94	91.06
	Tex-SAGE	90.79	44.41	92.17

Table 3: Overview of model accuracy on e-SNLI, ECQA and ComVE datasets. G-Tex achieves results that are better or comparable to the baselines. The best performance of each evaluation metric across all models is highlighted in bold.

Notably, the model incorporating GNNs only has approximately up to 0.28% more parameters than the baseline models T5 and 0.24% more parameters than the baseline models BART. Overall, the training time for different methods varies by only a few seconds.

Method	$Param_{T5}$	$Param_{BART}$	$Time_{T5}$
Fine-tuning	737M	406M	13:51
Prompt	737M	406M	14:23
Tex-GCN	738M	407M	13:41
Tex-GAT	738.1M	407M	13:42
Tex-SAGE	739.1M	407M	13:49

Table 4: Number of parameters and training time for different methods using T5 and BART.

H Faithfulness Evaluation Method

Following (Atanasova et al., 2023), we conduct the counterfactual evaluation to assess the faithfulness of the generated NLEs. Specifically, given an input instance x with the model's original answer y_0 and its corresponding NLE tokens $[y_1, \ldots, y_l]$ (see §3.1), we insert a word x_c into x, forming a new input x'. To ensure the coherence of x', we only insert random adjectives before nouns. For each original input x, we generate candidate insertions at 4 random positions, with 4 candidates per position, resulting in 16 perturbed inputs x' for each instance. If the model's prediction changes ($y'_0 \neq y_0$), the newly generated NLE should include the inserted word, i.e., $x_c \in [y'_1, \ldots, y'_{p+q}]$; otherwise, the original NLE is unfaithful as it is potentially misaligned with the model's reasoning. Note that the unchanged label provides no relevant information about the faithfulness of the NLE.

I Overall Explanation Evaluation Results on e-SNLI and ComVE Dataset of G-Tex using Tex-GAT and Tex-GCN

As shown in Table 5, we also report the results of our models **G-Tex** using **Tex-GAT** and **Tex-GCN**.

Regarding faithfulness, almost all of our models outperform all the baseline models on both datasets, achieving improvements of up to 17.18% with the T5-based **Tex-GCN** on the ComVE dataset, which demonstrates our approach's effectiveness in enhancing the faithfulness of NLEs.

Across different highlight explanation types, ***token interactive explanations*** consistently achieve the best faithfulness results on the e-SNLI dataset, regardless of the base model architecture. In contrast, on the ComVE dataset, ***highlight token explanations*** consistently demonstrate the highest faithfulness, highlighting the influence of dataset characteristics on the advantages of different explanation types in enhancing NLE faithfulness. For example, on the ComVE dataset, where the first part of the input is a general question in which the statement of the two is against comment sense, the simple interaction between the tokens/spans from the question and the statements might be less informative than simply selecting the important tokens from the statements. **This suggests that the choice of highlight explanation types to enhance NLE quality, particularly in terms of faithfulness, should be carefully tailored to the specific characteristics of the dataset.**

Regarding the similarity between the generated NLEs and the golden ones, as measured by automatic metrics, all the NLEs generated by our method on both datasets achieve equal or higher performance than the baselines. Among the different highlight explanation types, NLEs guided by ***highlight token explanations*** most frequently

achieve the highest similarity with the golden ones, both lexically and semantically.

Among the different GNN variants of our **G-Tex** method, **Tex-GAT**, **Tex-GCN**, and **Tex-SAGE**, there is no consistent trend indicating that any particular GNN layer consistently outperforms the others in improving the faithfulness or the similarity of the NLEs to the golden explanations.

J Statistical Uncertainty Measurement for Faithfulness Evaluation on e-SNLI and ComVE Datasets using Tex-SAGE and Fine-tuning$_{base}$ with T5-large and BART-large models

To demonstrate the significant improvement of our **G-Tex** in terms of faithfulness, we compute the p-values (Wasserstein and Lazar, 2016) for *Counter Unfaith* and *Total Unfaith* (see Section §5.1) when comparing the **Fine-tuning$_{base}$** and our **Tex-SAGE** model on the e-SNLI and ComVE datasets, using T5-large and BART-large with 5 random seeds.

As shown in Table 6, all p-values are less than 0.05, indicating that the natural language explanations generated by our **G-Tex** exhibit significantly lower unfaithfulness compared to the baseline method.

K Overall Explanation Evaluation Results on ECQA dataset for G-Tex based on T5-large and BART-large

K.1 Overall Explanation Evaluation Results on ECQA dataset for G-Tex based on T5-large

The faithfulness and automatic evaluation results of T5-based models on the ECQA dataset are shown in Table 7.

Regarding the faithfulness of NLEs, almost all of our methods outperform the baseline methods, highlighting the effectiveness of our framework. Among the different highlight explanation types, ***token interactive explanations*** demonstrate the best performance in generating faithful NLEs when using **Tex-GCN**, achieving 21.18% total unfaithfulness. Other variants, such as **Tex-GAT** and **Tex-SAGE**, also achieve comparable performance, with 21.44% and 21.74% total unfaithfulness, respectively. **On the ECQA dataset, *token interactive explanations* show a clear advantage over other highlight explanation types in improving the faithfulness of NLEs**.

Regarding the similarity between the generated NLEs and the gold ones, G-Tex outperforms the fine-tuning baseline in most settings. Although the prompt baseline achieves the highest SacreBLEU and BERTScore, G-Tex lags behind by only 1.537 in SacreBLEU and 0.004 in BERTScore. Among all types of highlight explanations, ***span interactive explanations*** achieve the highest scores with G-Tex.

K.2 Automatic Evaluation Results on ECQA dataset for G-Tex based on BART-large

As shown in Table 8, we also conduct automatic evaluation on BART-based G-Tex on ECQA datasets regarding Lexical and Semantical Similarity with golden explanations.

Compared to all the baseline methods, on ECQA dataset, with the highest scores always belong to our *token interactive explanation* guided **Tex-GCN** method, and other variants are with comparable performance to the baselines, our model also shows advantage in both lexical and semantic similarity.

Among the different explanation types, *token interactive explanations* demonstrate superior performance in both lexical and semantic metrics. Notably, ***token interactive explanations* show a slight advantage over the other two explanation types in generating NLEs with more plausible meanings to humans.**

K.3 Faithfulness Evaluation Results on ECQA dataset for G-Tex based on BART-large

We also evaluated the faithfulness of G-Tex based on BART-large on the ECQA dataset and observed that the faithfulness scores for all methods (including the baselines) were uniformly 100%. This result indicates that the BART-based models are prone to counterfactual attacks and none of these explanations were faithful. We attribute this outcome to the inherent complexity of the ECQA dataset and the potential vulnerability of the BART model to counterfactual attacks.

L Supplementary Automatic Explanation Evaluation Results for G-Tex based on T5-large and BART-large

To evaluate the similarity between the generated NLE and the golden ones as an approximation of plausibility to humans, we also leverage the following four metrics to evaluate their lexical and semantic similarity:

Explanation Type	Model	e-SNLI				ComVE			
		Unfaithfulness(%↓)		Automatic(↑)		Unfaithfulness(%↓)		Automatic(↑)	
		Counter	Total	SacreBLEU	BERTScore	Counter	Total	SacreBLEU	BERTScore
T5-based									
-	**Fine-tuning$_{base}$**	47.08	16.89	15.430	0.894	**87.17**	73.73	7.634	0.876
Highlight Token	**Prompt**	42.04	14.11	15.686	0.898	87.04	74.18	7.592	0.876
	Tex-GAT (Ours)	35.92	11.28	16.106	**0.899**	91.75	57.51	**8.990**	0.883
	Tex-GCN (Ours)	35.47	10.88	16.111	**0.899**	92.13	**57.00**	8.672	0.881
Token Interactions	**Prompt**	51.56	19.2	15.478	0.898	87.49	76.43	7.028	0.888
	Tex-GAT (Ours)	**34.28**	10.67	16.106	**0.899**	92.04	74.60	7.692	**0.891**
	Tex-GCN (Ours)	32.59	**10.03**	16.121	**0.899**	92.75	77.03	7.831	**0.891**
Span Interactions	**Prompt**	42.47	13.65	16.031	**0.899**	89.34	79.44	7.226	0.815
	Tex-GAT (Ours)	38.05	12.05	16.119	**0.899**	92.73	68.15	7.256	0.815
	Tex-GCN (Ours)	34.31	10.82	**16.160**	0.898	91.99	71.77	7.771	**0.891**
BART-based									
-	**Fine-tuning$_{base}$**	57.98	19.64	15.732	0.906	**82.72**	72.82	10.070	**0.891**
Highlight Token	**Prompt**	56.65	24.20	15.678	0.898	84.74	61.97	10.012	**0.891**
	Tex-GAT (Ours)	43.85	13.78	**16.503**	**0.909**	91.97	**58.11**	10.092	**0.891**
	Tex-GCN (Ours)	44.68	14.32	16.364	**0.909**	90.95	59.13	**10.489**	0.893
Token Interactions	**Prompt**	51.56	19.20	15.478	0.898	95.85	69.86	7.868	0.890
	Tex-GAT (Ours)	48.38	16.07	16.24	0.908	95.21	72.52	7.405	0.888
	Tex-GCN (Ours)	**41.57**	**12.89**	16.364	**0.909**	94.11	72.03	7.700	0.889
Span Interactions	**Prompt**	51.10	17.41	16.046	0.888	94.89	65.52	7.333	0.888
	Tex-GAT (Ours)	42.90	12.92	16.449	**0.909**	93.98	61.39	7.795	0.890
	Tex-GCN (Ours)	45.48	14.10	16.447	**0.909**	71.07	96.44	7.518	0.887

Table 5: Overall evaluation results on e-SNLI and ComVE datasets for T5-based and BART-based models, with our **G-Tex** model using **Tex-GAT** and **Tex-GCN**. Counter indicates *Counter Unfaith*, Total indicates *Total Unfaith*. The best performance of each evaluation metric is in bold. See Table 1 for results of our model using **Tex-SAGE**.

Explanation Type	Model	e-SNLI (P-Value)		ComVE (P-Value)	
		Counter Unfaith	Total Unfaith	Counter Unfaith	Total Unfaith
T5-based					
Highlight Token	**Tex-SAGE**	0.0007	0.0054	0.0136	0.0002
Token Interactions	**Tex-SAGE**	0.0002	0.0001	0.0164	0.0047
Span Interactions	**Tex-SAGE**	0.0010	0.0032	0.0001	0.0307
BART-based					
Highlight Token	**Tex-SAGE**	0.0067	0.0064	0.0455	0.0001
Token Interactions	**Tex-SAGE**	0.0122	0.0007	0.0168	0.0169
Span Interactions	**Tex-SAGE**	0.0033	0.0006	0.0403	0.0116

Table 6: P-values of our **Tex-SAGE** model compared to **Fine-tuning$_{base}$** on the e-SNLI and ComVE datasets, using T5-large and BART-large, regarding *Counter Unfaith* and *Total Unfaith* on 5 random seeds.

Rouge1 (Lin, 2004) calculates the overlap of unigrams between the generated explanation and the golden ones, providing insight into lexical similarity at the word level.

RougeL (Lin, 2004) measures the longest common subsequence between the generated explanation and the golden explanations.

MoverScore (Zhao et al., 2019) calculates semantic similarity by computing word embeddings and their movement cost, capturing meaning while accounting for variations in word order and structure.

BARTScore (Yuan et al., 2021) leverages BART's language model to assess the likelihood of the reference text being generated given the generated explanation as input, providing a fluency and relevance measure.

L.1 Supplementary Automatic Explanation Evaluation Results for G-Tex based on T5-large

As shown in Table 9, Table 10 and Table 11, we conduct a supplementary automatic evaluation on T5-based G-Tex regarding Lexical Similarity and Semantic Similarity with the golden explanations on e-SNLI, ECQA and ComVE datasets respectively.

Compared to all the baseline methods on the e-SNLI dataset, **all variants of our G-Tex achieve higher lexical and semantic similarity with gold explanations**, indicating that our approach can generate more plausible NLEs. For instance, we observe up to a 2.1% improvement in ROUGE-1 and a notable absolute increase of 0.224 in BARTScore. On the ECQA dataset, our G-Tex achieves better similarity performance than Fine-tuning$_{base}$

Evaluation Metrics		UnFaithfulness(% ↓)		Automatic Evaluation (↑)	
		Counter Unfaith	Total Unfaith	SacreBLEU (0-100)	BERTScore (0-1)
Fine-tuning$_{base}$		49.34	24.80	14.057	0.883
Highlight Tokens	**Prompt**	46.56	25.27	15.303	0.887
	Tex-GAT	44.76	21.99	14.048	0.883
	Tex-GCN	49.61	25.21	13.855	0.882
	Tex-SAGE	45.42	22.44	13.968	0.882
Token Interactions	**Prompt**	51.29	33.30	15.311	0.887
	Tex-GAT	43.49	21.44	13.910	0.882
	Tex-GCN	**43.42**	**21.18**	14.079	0.883
	Tex-SAGE	44.20	21.74	13.978	0.882
Span Interactions	**Prompt**	50.20	28.22	**16.046**	**0.888**
	Tex-GAT	49.22	23.85	14.339	0.883
	Tex-GCN	50.46	24.91	14.477	0.883
	Tex-SAGE	46.87	22.50	14.509	0.884

Table 7: Overall Evaluation Results on ECQA of T5-based G-Tex. The best performance of each evaluation metric across all NLE generation models is in bold.

Automatic Evaluation Metrics		Lexical Similarity (↑)			Semantic Similarity (↑)		
		ROUGE-1 (0-1)	ROUGE-L (0-1)	SacreBLEU (1-100)	MoverScore (0-1)	BARTScore (-0-1)	BERTScore (0-1)
Fine-tuning$_{base}$		0.180	0.130	12.484	0.840	-4.433	0.836
Highlight Tokens	**Prompt**	0.112	0.077	10.733	0.767	-4.557	0.754
	Tex-GAT (Ours)	0.172	0.125	12.186	0.837	-4.453	0.835
	Tex-GCN (Ours)	0.198	0.146	13.091	0.840	-4.379	0.839
	Tex-SAGE (Ours)	0.181	0.133	12.659	0.839	-4.434	0.836
Token Interactions	**Prompt**	0.185	0.134	12.724	0.838	-4.435	0.837
	Tex-GAT (Ours)	0.208	0.151	13.519	0.841	-4.399	0.841
	Tex-GCN (Ours)	**0.321**	**0.226**	**17.860**	**0.848**	**-4.079**	**0.858**
	Tex-SAGE (Ours)	0.243	0.174	14.773	0.843	-4.269	0.847
pan Interactions	**Prompt**	0.175	0.126	12.288	0.839	-4.454	0.835
	Tex-GAT (Ours)	0.176	0.128	12.295	0.838	-4.456	0.835
	Tex-GCN (Ours)	0.175	0.128	12.364	0.838	-4.455	0.835
	Tex-SAGE (Ours)	0.186	0.135	12.802	0.839	-4.415	0.837

Table 8: Automatic Evaluation Results on ECQA of BART-based G-Tex. The best performance of each evaluation metric across different NLE generation models is in bold.

(which does not utilize explanation information) and is comparable to the prompt-based baseline. On the ComVE dataset, all NLEs generated by our method incorporating ***highlight token explanations*** surpass the baselines in both lexical and semantic similarity, while the variants based on ***token interactive explanations*** and ***span interactive explanations*** sometimes fail to do so. This is likely due to the format of the ComVE dataset, which presents a simple question followed by two similar statements. In this scenario, ***token interactive explanations*** and ***span interactive explanations*** may struggle to capture sufficient information from the limited interaction between the question and the options.

Among the different highlight explanation types on the e-SNLI dataset, ***token interactive explanations***, particularly those using the Tex-SAGE variant of our G-Tex, achieve the highest lexical and semantic similarity. Meanwhile, ***highlight token explanations*** and ***span interactive explanations*** also perform strongly, excelling at ROUGE-L and ROUGE-1 scores respectively. On the ECQA dataset, ***span interactive explanations*** have a slight edge over other explanation types, although the difference is marginal. On the ComVE dataset, ***highlight token explanations*** show a clear advantage across all metrics. This is likely due to the input format of the ComVE dataset, which makes it challenging for ***token interactive explanations*** and ***span interactive explanations*** to capture sufficient information, as discussed earlier.

In summary, **these findings highlight that the advantages of different explanation types in improving NLE quality vary with dataset characteristics.**

Automatic Evaluation Metrics		Lexical Similarity(↑)		Semantic Similarity(↑)	
		ROUGE-1 (0-1)	ROUGE-L (0-1)	MoverScore (0-1)	BARTScore (-0-1)
Fine-tuning$_{base}$		0.448	0.384	0.838	-3.646
Highlight Tokens	**Prompt**	0.455	0.397	0.840	-3.492
	Tex-GAT (Ours)	0.467	0.402	0.842	-3.437
	Tex-GCN (Ours)	0.468	0.403	0.842	-3.425
	Tex-SAGE (Ours)	0.468	**0.404**	0.841	**-3.422**
Token Interactions	**Prompt**	0.459	0.394	0.842	-3.503
	Tex-GAT (Ours)	0.467	0.402	0.842	-3.437
	Tex-GCN (Ours)	0.467	0.403	0.842	-3.435
	Tex-SAGE (Ours)	**0.469**	**0.404**	**0.843**	-3.431
Span Interactions	**Prompt**	0.466	0.402	0.841	-3.467
	Tex-GAT (Ours)	0.466	0.403	0.841	-3.442
	Tex-GCN (Ours)	**0.469**	0.403	0.843	-3.433
	Tex-SAGE (Ours)	0.467	0.402	0.842	-3.428

Table 9: Automatic Evaluation Results on e-SNLI of T5-based G-Tex (excluding SacreBLEU and BERTScore, which are presented in Table 1). The best performance of each evaluation metric across different NLE generation models is in bold.

L.2 Supplementary Automatic Explanation Evaluation Results for G-Tex based on BART-large

As shown in Table 12, Table 8 and Table 13, we conduct a supplementary automatic evaluation on BART-based G-Tex regarding Lexical Similarity and Semantic Similarity with the golden explanations on e-SNLI, ECQA and ComVE datasets respectively.

M Human Evaluation

In line with prior work (Atanasova et al., 2020b; Jolly et al., 2022), our human evaluation assesses the generated explanations across four key dimensions:

Coverage: The explanation includes all important and salient information, ensuring no significant points that contribute to label prediction are omitted.

Non-redundancy: The explanation should avoid redundant, repeated, or irrelevant information and should not include content that is unreasonable or inconsistent with common sense.

Non-contradiction: The explanation should not contradict the predicted label or the input text, maintaining consistency throughout.

Overall Quality: The explanations are rated based on overall quality, considering factors such as grammar, readability, and clarity.

We engaged three PhD students with backgrounds in computer science to evaluate the explanations using a 1–7 Likert scale following previous work (Castro Ferreira et al., 2019; Ribeiro et al., 2021; Yuan and Färber, 2024). We compare the text generated by the Fine-tuning$_{base}$ with that generated by **Tex-GAT** when guided by *highlight token*, *token interactive explanations*, and *span interactive explanations*, respectively. The annotator agreement is reported in Table 17. Note that we randomly sample 100 NLEs generated by each model.

M.1 Human Evaluation Results

e-SNLI In Table 14, across all highlight explanation types, the NLEs generated by the *token interactive explanations* achieve the highest scores across most dimensions, particularly excelling in *Non-redundancy* (5.95) and *Overall Quality* (6.37), indicating its effectiveness in producing concise and high-quality explanations. The NLEs generated with the guidance of *span interactive explanations* method also show strong performance, especially in *Non-contradiction* (6.72), suggesting that modeling span-level interactions is beneficial for maintaining consistency of the NLE with the generated label. The *highlighted token explanations* performs slightly lower, indicating that while it captures key tokens effectively, it may miss out on broader contextual relationships crucial for non-redundancy and overall quality.

ECQA Table 15 shows the evaluation results for the ECQA dataset, where the NLEs generated by *token interactive explanations* again lead in *Non-redundancy* (4.82) and achieves a high *Non-contradiction* score (5.08), confirming its robust-

Automatic Evaluation Metrics		Lexical Similarity(↑)		Semantic Similarity(↑)	
		ROUGE-1 (0-1)	ROUGE-L (0-1)	MoverScore (0-1)	BARTScore (-0-1)
Fine-tuning$_{base}$		0.469	0.346	0.850	-3.584
Highlight Tokens	**Prompt**	0.490	0.355	**0.857**	**-3.528**
	Tex-GAT (Ours)	0.469	0.346	0.851	-3.576
	Tex-GCN (Ours)	0.468	0.347	0.850	-3.575
	Tex-SAGE (Ours)	0.468	0.347	0.850	-3.569
Token Interactions	**Prompt**	0.489	0.354	0.855	-3.549
	Tex-GAT (Ours)	0.468	0.345	0.849	-3.598
	Tex-GCN (Ours)	0.469	0.346	0.850	-3.593
	Tex-SAGE (Ours)	0.468	0.346	0.851	-3.593
Span Interactions	**Prompt**	**0.496**	**0.360**	**0.857**	-3.520
	Tex-GAT (Ours)	0.472	0.350	0.850	-3.569
	Tex-GCN (Ours)	0.470	0.349	0.849	-3.568
	Tex-SAGE (Ours)	0.474	0.350	0.851	-3.560

Table 10: Automatic Evaluation Results on ECQA of T5-based G-Tex (excluding SacreBLEU and BERTScore, which are presented in Table 7). The best performance of each evaluation metric across different NLE generation model is in bold.

ness across different datasets. The *span interactive explanations* perform similarly well, attaining the highest *Overall Quality* score (5.63), emphasizing its adaptability in varied datasets.

Overall, while the *highlight token explanations* shows slightly lower performance across all highlight explanation types, leveraging *span interactive explanations* and *token interactive explanations* that are encoded in G-Tex notably improves the quality and consistency of the generated explanations.

M.2 Human Evaluation Instruction

The annotators are asked to rate the generated texts following the instructions in Table 16.

M.3 Pairwise agreement for human annotations

Table 17 shows Pairwise agreement for human annotations for NLE generated by T5-based G-Tex on e-SNLI and ECQA dataset.

Automatic Evaluation Metrics		Lexical Similarity(↑)		Semantic Similarity(↑)	
		ROUGE-1 (0-1)	ROUGE-L (0-1)	MoverScore (0-1)	BARTScore (-0-1)
Fine-tuning$_{base}$		0.355	0.319	0.828	-4.030
Highlight Tokens	**Prompt**	0.354	0.317	0.825	-4.051
	Tex-GAT (Ours)	**0.394**	0.332	0.832	-3.884
	Tex-GCN (Ours)	0.384	**0.333**	0.830	-3.934
	Tex-SAGE (Ours)	0.393	0.330	**0.833**	**-3.881**
Token Interactions	**Prompt**	0.312	0.269	0.817	-4.083
	Tex-GAT (Ours)	0.326	0.283	0.816	-3.976
	Tex-GCN (Ours)	0.332	0.288	0.817	-3.970
	Tex-SAGE (Ours)	0.310	0.266	0.817	-4.070
Span Interactions	**Prompt**	0.317	0.275	0.815	-4.059
	Tex-GAT (Ours)	0.324	0.280	0.815	-3.998
	Tex-GCN (Ours)	0.328	0.286	0.815	-3.975
	Tex-SAGE (Ours)	0.328	0.283	0.818	-3.980

Table 11: Automatic Evaluation Results on ComVE of T5-based G-Tex (excluding SacreBLEU and BERTScore, which are presented in Table 1). The best performance of each evaluation metric across different NLE generation models is in bold.

Automatic Evaluation Metrics		Lexical Similarity(↑)		Semantic Similarity(↑)	
		ROUGE-1 (0-1)	ROUGE-L (0-1)	MoverScore (0-1)	BARTScore (-0-1)
Fine-tuning$_{base}$		0.457	0.391	0.838	-3.491
Highlight Tokens	**Prompt**	0.468	0.398	**0.843**	-3.458
	Tex-GAT (Ours)	0.476	0.405	**0.843**	**-3.403**
	Tex-GCN (Ours)	0.474	0.402	0.841	-3.415
	Tex-SAGE (Ours)	0.474	0.402	0.840	-3.416
Token Interactions	**Prompt**	0.459	0.394	**0.843**	-3.503
	Tex-GAT (Ours)	0.472	0.401	0.841	-3.449
	Tex-GCN (Ours)	0.473	0.402	0.842	-3.418
	Tex-SAGE (Ours)	0.472	0.403	0.841	-3.431
Span Interactions	**Prompt**	0.475	0.403	0.841	-3.419
	Tex-GAT (Ours)	**0.477**	0.403	0.842	-3.427
	Tex-GCN (Ours)	0.476	0.403	0.842	-3.423
	Tex-SAGE (Ours)	**0.477**	**0.404**	0.842	-3.423

Table 12: Automatic Evaluation Results on e-SNLI of BART-based G-Tex (SacreBLEU and BERTScore are excluded and are presented in Table 1). The best performance of each evaluation metric across different NLE generation models is in bold.

Automatic Evaluation Metrics		Lexical Similarity(↑)		Semantic Similarity(↑)	
		ROUGE-1 (0-1)	ROUGE-L (0-1)	MoverScore (0-1)	BARTScore (-0-1)
Fine-tuning$_{base}$		0.421	0.325	**0.840**	-3.802
Highlight Tokens	**Prompt**	0.419	0.322	0.834	-3.796
	Tex-GAT (Ours)	0.427	0.325	0.837	-3.765
	Tex-GCN (Ours)	**0.435**	**0.332**	0.838	-3.761
	Tex-SAGE (Ours)	0.434	0.330	0.837	**-3.748**
Token Interactions	**Prompt**	0.334	0.284	0.818	-4.036
	Tex-GAT (Ours)	0.322	0.277	0.818	-4.047
	Tex-GCN (Ours)	0.334	0.285	0.817	-3.985
	Tex-SAGE (Ours)	0.316	0.269	0.814	-4.129
Span Interactions	**Prompt**	0.323	0.274	0.818	-4.029
	Tex-GAT (Ours)	0.334	0.288	0.818	-4.011
	Tex-GCN (Ours)	0.327	0.278	0.818	-4.045
	Tex-SAGE (Ours)	0.333	0.287	0.820	-4.017

Table 13: Automatic Evaluation Results on ComVE of BART-based G-Tex. The best performance of each evaluation metric across different NLE generation models is in bold.

Method	Coverage	Non Redund.	Non Contrad.	Overall
Fine-tuning$_{base}$	6.72	5.86	6.67	6.28
Highlight Tokens	6.74	5.80	6.67	6.06
Token Interactions	**6.75**	**5.95**	6.64	**6.37**
Span Interactions	6.67	5.92	**6.72**	6.26

Table 14: Human Evaluation Results on e-SNLI dataset of our **G-Tex** using **Tex-GAT** based on T5.

Method	Coverage	Non Redund.	Non Contrad.	Overall
Fine-tuning$_{base}$	5.66	4.41	4.91	5.53
Highlight Tokens	5.08	4.27	4.51	5.20
Token Interactions	5.60	**4.82**	**5.08**	5.61
Span Interactions	**5.65**	4.67	4.90	**5.63**

Table 15: Human Evaluation Results on ECQA dataset of our **G-Tex** using **Tex-GAT** based on T5.

Criterion and Explanation	1 - 3 (Very Bad)	3 - 5 (OK, but not good enough)	5 - 7 (Good to Very Good)
Coverage: The explanation contains important, salient information and does not miss any important points that contribute to the label prediction.	The explanation misses the most critical points in the input text.	The explanation provides a reason for the prediction, but not the main reason.	The explanation covers the most important points/reasons for the prediction.
Non-redundancy: The explanation does not contain any information that is redundant, repeated, or irrelevant to the claim and predicted label. It should also be reasonable according to common sense.	The explanation contains irrelevant information, unnecessary repetition, or elements that do not appear in the input text; violates common sense.	The explanation is acceptable but contains some redundancy or repetition.	Slightly to no redundancy, repetition, or hallucination.
Non-contradiction: The explanation does not contain any pieces of information that are contradictory to the predicted label and the input text.	The explanation contradicts the predicted label or input text; they address different topics.	The explanation matches the predicted label but is not fully logical.	The explanation and predicted label are fully consistent and logical.
Overall Quality: Rank the explanations by their overall quality. Consider grammar, readability, and clarity.	Many grammatical errors, difficult to understand.	No major grammar mistakes, but not easy to understand.	Perfect grammar and language clarity.

Table 16: Rating Criteria for Generated Natural Language Explanations

-	Coverage		Non-redundancy		Non-contradiction		Overall	
Annotator_id	**2**	**3**	**2**	**3**	**2**	**3**	**2**	**3**
				e-SNLI				
1	0.51	0.25	0.53	0.43	0.36	0.19	0.33	0.16
2	-	0.40	-	0.53	-	0.43	-	0.37
Mean	0.39		0.49		0.33		0.29	
				ECQA				
1	0.35	0.20	0.33	0.15	0.58	0.40	0.27	-0.02
2	-	0.10	-	0.29	-	0.35	-	0.30
Mean	0.22		0.26		0.44		0.18	

Table 17: Pairwise agreement for human annotations on e-SNLI and ECQA. We report separately the agreement between annotator pairs 1-2, 2-3, and 1-3. Mean represents the average over three pairwise agreements.