AUTOMATIC SCIENTIFIC CLAIMS VERIFICATION WITH PRUNED EVIDENCE GRAPH

Liri Fang, Dongqi Fu, Vetle I. Torvik
University of Illinois Urbana-Champaign, Meta AI
{lirif2, vtorvik}@illinois.edu, {dongqifu}@meta.com

ABSTRACT

In general, *automatic scientific claim verification* methods retrieve evidence paragraphs, select rationale sentences, and predict the sentence stances for a given claim. Large language models (LLMs) are expected to be the next-generation tool to solve this task. However, due to the domain-specific claims, LLMs trained on the large-scale general corpus at least need external knowledge to warm up. Therefore, how to extract qualified and reasonable sentences with their stances toward a given claim is indispensable. GraphRAG is designed to learn the hierarchical relationships of context and selectively retrieve related information, improving LLMs' reasoning in ad-hoc and domain-specific claim verification scenarios. Nevertheless, current GraphRAG methods typically require a pre-existing domain-specific knowledge base. Hence, a natural question can be asked: *How far are we from automatically building a semantic graph and selecting rationale sentences for a pre-trained LLM, and which process is better to be independent of the pre-trained LLM?*

In this paper, we propose our ongoing research on distilling information across sentences by constructing a complete evidence graph and pruning it to capture the relevant connections between claim and paragraph sentences. This enables updating the sentence embeddings, and consequently enhances multiple-rationale sentence identification and stance prediction. The effectiveness of this proposed framework is empirically tested on SciFact, i.e., an open-access dataset in the biomedical domain. From the current stage, we discern that selected baselines, including our method, can hardly outperform across all experimental settings, which indicates many future research directions for researchers and practitioners.

1 INTRODUCTION

Scientific claim verification tends to focus on findings reported in the scientific literature as opposed to countering disinformation in social media contents (Wadden et al., 2020). With the rapid growth of scientific research, it becomes more and more challenging for researchers to keep up with the latest scientific outcomes, and retrieve and verify the literature corresponding to specific claims. *Automatic* scientific claim verification (Wadden et al., 2020; Liu et al., 2020; Zhong et al., 2020; Li et al., 2021; Wadden et al., 2022; Alvarez et al., 2024; Vladika et al., 2025) is proposed for numerous applications, like scientific paper recommendations, scientific question answering, etc.

As shown in Figure 1, the general framework for automatic claim verification includes three modules: (1) a module to select evidence paragraph for a given claim; (2) a module to select rationale sentences from the paragraph for a given claim; and (3) a classifier to predict the stances of the rationale sentences to the given claim. Under this pipeline, most existing approaches organize the claim and scientific document as a sequence of tokens and utilize Transformer-based pre-trained language models (PLMs) in the pipeline or joint training fashion (Wadden et al., 2020; DeYoung et al., 2020; Devlin et al., 2019; Zhang et al., 2021). Two challenges originate from this simple sequence document modeling: (1) one evidence scientific document may consist of multiple rationale sentences, supporting or contradicting the claim; (2) these rationale sentences may be scattered across paragraphs. For example, two rationale sentences in Figure 2 locate in the first and fifth sentences of the evidence paragraph. These two challenges are related to the complex structure of scientific documents that are formatted following standard structures and can be segmented into rhetorical categories in order (Mann & Thompson, 1987; Agarwal & Yu, 2009).

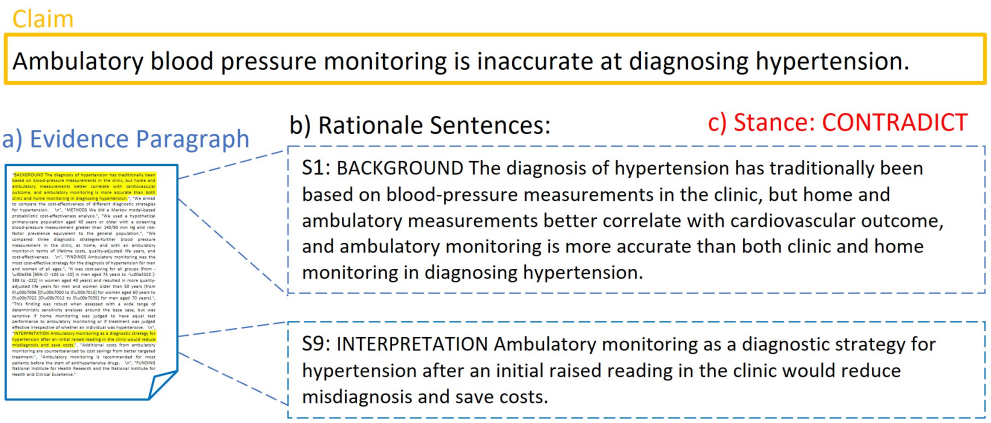

Figure 1: An example claim from the SciFact dataset. The general framework for claim verification includes three modules: a) a module to select evidence paragraph for a given claim, b) a module to select rationale sentences from the paragraph for a given claim, and c) a classification module to predict stances of the rationale sentences to the given claim.

The above analysis calls for an in-depth investigation of structures for scientific claim verification from at least two aspects.

- First, facing complex underlying semantic structures in scientific documents, nascent studies propose to use graph modeling for the input document, whose assumption is based on the fact that scientific documents consist of sets of sentences and hierarchical sets of sentences that are related to each other (Mann & Thompson, 1987). For example, graph neural network modeling (Fu & He, 2021; Zhou et al., 2022; Fu et al., 2022b; 2024b; Tieu et al., 2024; Zheng et al., 2024) associated with the construction of complete evidence graphs can be used to capture complex relational and logical information of natural language (Thorne et al., 2018; Liu et al., 2020; Fu et al., 2022a).
- Second, large language models (LLMs) are usually trained on the massive general text corpus (Zhao et al., 2023), which may be insufficient to answer domain-specific scientific questions (e.g., biomedical claim and drug discovery) but relies on the involvement of external knowledge, like fine-tuning(Zhang et al., 2023; Han et al., 2024), in-context learning (Dong et al., 2024; Sahoo et al., 2024), and retrieval-augmented generation (RAG) (Gao et al., 2023; Fan et al., 2024). To be specific, external knowledge is also structurally organized, which requires the corresponding method to pay attention to hierarchical representation and multi-hop reasoning and prompt the GraphRAG research direction (Edge et al., 2024).

The aforementioned two aspects reflect two possible research directions for automatic scientific claim verification. The first direction depends on developing the large-scale graph foundation models (Li et al., 2024b; Liu et al., 2025; Zhu et al., 2025). The second direction is expecting to develop lightweight graph representation learning methods to select multi-hop evidence for enriching the knowledge of pre-trained LLMs (Grattafiori et al., 2024; Guo et al., 2025; Fu et al., 2024a) via instruction-tuning or prompting methods.

Following the second direction, in this paper, we are curious about "***in which part and how graph representation learning methods can help LLMs verify domain-specific scientific claims?***" Motivated by this question, we present our preliminary studies in this paper.

First of all, the recent GraphRAG method (Li et al., 2024a) relies on the existing knowledge base as the input graph (Li et al., 2025), such as the Wikipedia knowledge graph. However, for an ad-hoc and specific domain, a previously deliberately built knowledge base is not quite possible and not easy to acquire; it requires users to build a structural knowledge base at the first step (Zou et al., 2025). For the scientific claim verification, we refer to this knowledge graph as an evidence graph (e.g., the node can be the stance of a sentence towards a given claim, and the formal dedication and expression can be found in Section 3) in the rest of the paper.

A naive solution is to build a complete (or fully connected) evidence graph. Then, users can use this graph to select if a sentence is rational to a given claim and what is the sentence's stance (e.g., support or refute), as shown in Section 3, to provide external useful information for LLM reasoning via in-context learning, etc.

However, a complete evidence graph captures contextualized information from all sentences in the paragraph, but does not necessarily guarantee distinguishing the order or distance of those sentences. Furthermore, a complete evidence graph may not represent the underlying structure of scientific documents and may introduce noisy connections when reasoning across the graph.

To serve as an auxiliary tool for enriching LLM's external knowledge, besides the graph representation learning ability, we also expect the proposed technology can avoid the overhead of LLM's usage cost, just learn the basic semantics meaning by small pre-trained language models (PLMs), and focus more on the structure discovery of external knowledge.

Hence, in this paper, we propose our ongoing framework **PrunE** that learns to Prune the sentence-level complete Evidence graph with positional encoding for the scientific claim verification task. To be specific, PrunE first extends the complete evidence graph with the positional encoding of sentences without losing the sequential information. Then, PrunE learns to prune the complete evidence graph using binary gates in order to mine task-relevant connections.

2 Related Work

The current proposed frameworks (Wadden et al., 2020; DeYoung et al., 2020; Glockner et al., 2020; Li et al., 2021; Liu et al., 2020; Zhong et al., 2020; Zhou et al., 2019) have similar pipelines, including a) a model to retrieve evidence documents(abstracts) for a given claim, b) a model to select rationale sentences for a given claim, and c) a classification model to predict stances of the rationale to the given claim.

2.1 Pretrained LMs for Claim Verification

Most existing approaches to automatic scientific claim verification leverage pretrained language models (PLMs) such as BERT and SciBERT as sentence encoders and classifiers. These models typically process concatenated claim-evidence pairs independently for evidence retrieval and stance prediction (Wadden et al., 2020; DeYoung et al., 2020; Glockner et al., 2020). PLMs can model intra-pair relationships via self-attention mechanisms.

2.2 LLM-based Scientific Claim Verification

More recently, large language models (LLMs) have been explored for scientific claim verification in zero- and few-shot settings. Alvarez et al. (2024) propose a framework in which LLMs generate negated variants of scientific claims to serve as negative training examples. They use in-context learning (ICL) for stance inference and introduce Scitance, a dataset built upon SciFact that includes citation-level annotations. Vladika et al. (2025) present a step-by-step verification pipeline for medical claims, incorporating explainable reasoning through chain-of-thought (CoT) prompting. Their system generates clarification questions, retrieves evidence from open-domain corpora such as PubMed and Wikipedia, and combines retrieved content with LLM internal knowledge to improve semantic understanding and verdict prediction. These works illustrate the potential of LLMs to generalize scientific reasoning across domains, though they also raise challenges related to grounding, hallucination, and controllability.

2.3 Graph-Based Scientific Claim Verification

Graph-based approaches have been increasingly explored for modeling inter-sentence and inter-entity relationships in scientific claim verification. These methods construct structured representations, such as sentence-level or semantic-level graphs, to facilitate more expressive reasoning. Zhou et al. (2019) construct a fully connected evidence graph, considering each rationale sentence as a node, and leverages GCN to convert the stance prediction task into a node classification task. Liu et al. (2020) extend this approach by introducing an edge kernel attention mechanism to capture richer

sentence-level dependencies via edge-based message passing. Zhong et al. (2020) introduce reasoning over a semantic-level graph for fact checking, which models higher-level semantic structures among sentence segments. Their method reorders sentences based on semantic relatedness and integrates them via graph-based representations to support more structured reasoning. More recently, Jeon & Lee (2025) propose GraphCheck, a multi-path fact-checking framework that uses entity-relationship graphs. Given a claim, their model generates knowledge triplets using LLM prompts and verifies them using document evidence.

2.4 Multi-Task Learning in Claim Verification

One normally-used training method is to train each model in the pipeline. However, this would cause the stance prediction model, which could worsen the performance more if the rationale sentence selection model, the former model, returns a wrong prediction. Therefore, jointly training the rationale sentence selection task and the stance prediction task as multi-task learning have been considered (Ma et al., 2018; Li et al., 2021). Li et al. (2021) use the cross-entropy loss as the objectives for both tasks and compute a weighted sum of two objectives. Also, Ma et al. (2018) build the model with task-shared and task-specific layers and add l-2 regularization to the sum of each tasks' objective.

2.5 Explainable Claim Verification

Various methods have been applied to the explainable claim verification task, including attention-based methods (Shu et al., 2019) and explanation generation (Kotonya & Toni, 2020). Shu et al. (2019) leverage co-attention networks over the news and its comments and visualize the explanation by outputting the last co-attention layer's weights. Kotonya & Toni (2020) mention that the scientific evidence would be difficult to understand; thus, they focus on generating abstractive explanations. This paper focuses on explaining the co-attention method, as natural language generation would be a different direction.

3 Problem Definition

According to (Wadden et al., 2020), automatic scientific claim verification can be expressed as follows. Given a claim c, a textual abstract $a \in \mathcal{A}$ has the stance label $y(c, a) \in \{$*SUPPORTS*, *REFUTES*, *NOINFO*$\}$, and only *SUPPORTS* and *REFUTES* are **evidence abstracts**. In each evidence abstract, not every sentence s_i, $i \in \{1, ..., |a|\}$, support or refute a given claim. Thus, those related sentences are defined as **rationale sentences**. For example, an evidence abstract a to claim c can have m rationale sentences $\{r_1(c, a), ..., r_m(c, a)\}$, and $m \leq |a|$.

To identify those rationale sentences through their latent relationships, a complete evidence graph $\mathcal{G} = (V, E)$ is proposed to be constructed for the evidence abstract a and its claim c (Thorne et al., 2018; Liu et al., 2020). To be specific, in graph $\mathcal{G}$, a node, consisting of a sentence s_i and the claim c pair, is denoted by (s_i, c) and $i \in |a|$. Therefore, the initial problem can be transferred to the node classification problem, i.e., given the complete graph structure and node features from (s_i, c), and m ground-truth rationale sentences should be classified out of all given sentences in an abstract.

However, the fully connected graph structure introduces noise and hinders scalability at the same time. Inspired by that, we aim to learn an evidence graph $\mathcal{G}$, which has both a sparse and meaningful layout to accurately predict:

- whether an abstract a is an evidence abstract to claim c?
- what is the exact label of abstract a (i.e., *SUPPORTS* or *REFUTES*)?
- what sentences in a are rationale sentences?

4 Methodology

PrunE consists of the following components, as illustrated in Figure 2: 1) a sentence encoder component that transform the token- and sentence- level textual sequence into the embedding space for a claim and paragraph sentences, 2) an evidence graph topology adaptation component that prunes

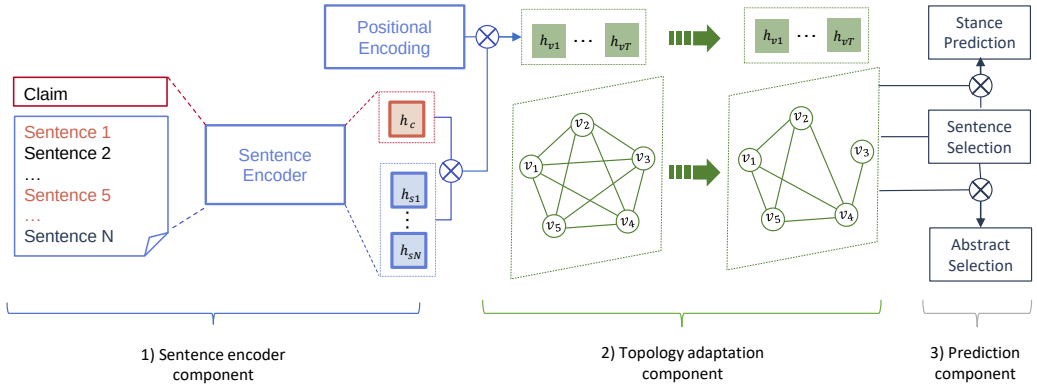

Figure 2: Overview of our proposed PrunE framework

the task-irrelevant edges of complete evidence graph and consolidates the embedding of evidence pairs accordingly. 3) a prediction component that is responsible for sentence identification, stance prediction, and abstract retrieval.

4.1 Sentence Encoder Component

The sentence encoder component maps the input textual sequence into the embedding space by leveraging PLMs. First, the claim and paragraph sentences $[c, s_1, .., s_N]$ are serialized and concatenated as the input sequence of PLMs. The first and last hidden states of PLMs are averaged as the token embeddings $\mathbf{h_t} \in \mathbb{R}^{m \times d}$. m is the number of tokens, and d is the output dimensions of PLMs. The token embeddings are fed into the pooling layer, attention mechanism (Bahdanau et al., 2015) to generate the sentence-level embeddings $\mathbf{h_s} = \sum_i (\alpha^i \mathbf{h}_t^i),\ \alpha^i = \text{softmax}(\mathbf{w}^{i\top} \mathbf{h}_t^i)$ for claim and paragraph sentences (Zhang et al., 2021; Li et al., 2021), where α^i is the learnable attention weights and $\mathbf{h}_t^i$ is the token embedding.

4.2 Evidence Graph Topology Adaptation Component

To model the complex structure of scientific documents, the evidence graph topology adaptation component first constructs a complete evidence graph with positional encoding, which preserves both contextualized information and positional information within paragraph sentences. Besides, the methods are introduced to learn how to prune the complete evidence graph to predict task-relevant sentence-level evidence graphs and consolidate sentence embeddings for the claim verification task.

4.2.1 Evidence Graph Construction

The evidence graph represents the content of each abstract a and the given claim c, capturing the relationships across the evidence node. In the complete evidence graph, each node represents an evidence pair $v_i = (c, s_i)$ of claim c and sentence s_i in a candidate abstract a. Mathematically, the evidence graph can be denoted as $\mathcal{G} = (V, E)$, where $V = \{v_i\}, E = \{(v_i, v_j), v_i \in V, v_j \in V\}$, and adjacency matrix $A \in \mathbb{R}^{N \times N}, N = |a|$. The simple concatenation of the evidence sentence and the claim may be insufficient to model the complex relationship between the claim sentence and the paragraph sentences. Therefore, we generate the node representations $\mathbf{h_{v_i}}$ by combining the claim sentence embedding $\mathbf{h_c}$ and paragraph sentence embeddings $\mathbf{h_{s_i}}$, as follows.

$$\mathbf{h_{v_i}} = [\mathbf{h_c} \odot \mathbf{h_{s_i}}, |\mathbf{h_c} - \mathbf{h_{s_i}}|, \mathbf{h_c}, \mathbf{h_{s_i}}, \mathbf{h_p}] \tag{1}$$

where $\odot$ denotes the element-wise multiplication. $\mathbf{h_p}$ denotes the positional encoding of each sentence in the paragraph (Vaswani et al., 2017).

4.2.2 TOPOLOGY ADAPTATION

The complete evidence graph could only provide redundant connections for the claim verification task. Here, PrunE extends the Graph Neural Networks (GNNs) model with trainable binary gates to learn the sparse topology of the complete evidence graph and generate enhanced node embeddings. In detail, we integrate the GNNs with hard concrete distribution for binary gates $Z \in \{0,1\}^{|V|\times|V|}$ to optimize the $\mathcal{L}_0$ regularization (Louizos et al., 2018; Luo et al., 2021). In this paper, Graph Attention Neural Networks (GAT) are used as the backbone GNN model.

Graph Attention Neural Network. The GAT layer implicitly weights the neighborhood nodes differently based on attention scores α_{v_i,v_j} (Velickovic et al., 2018).

$$\begin{aligned} h_{v_i}^{(l)} &= \sum_{v_j \in \mathcal{N}(v_i)} \alpha_{v_i,v_j} h_{v_j}^{(l-1)} \\ \alpha_{v_i,v_j} &= \text{softmax}(a^\top [W h_{v_i} \oplus W h_{v_j}]) \end{aligned} \tag{2}$$

Sparsification. Even though GAT can assign various weights to the neighbors, the task-irrelevant connections can still introduce noisy information for neighborhood aggregation (Luo et al., 2021). To prune these task-irrelevant connections, PrunE aims to learn a binary gate Z^l to filter out the task-irrelevant connections of the adjacency matrix and generate the new adjacency matrix $\tilde{A}^l = A \odot Z^l$ by penalizing the non-zero elements of the Z^l, for each layer l of GAT.

$$\mathcal{L}_0^l = \|Z^l\| = \sum_{v_i,v_j \in E} \mathbb{1}[z_{v_i,v_j}^l \neq 0] \tag{3}$$

In order to enable to differentiate the $\mathcal{L}_0$ loss with the respect of Z, the reparameterization trick of z_{v_i,v_j}^l is used, by introducing a Bernoulli distribution with parameter π_{v_i,v_j}^lover gate z_{v_i,v_j}^l, i.e., $q(z_{v_i,v_j}^l|\pi_{v_i,v_j}^l) = Bern(\pi_{v_i,v_j}^l)$. Accordingly, we can rewrite the attention scores α_{v_i,v_j} of the GAT layer as follows.

$$\tilde{\alpha}_{v_i,v_j} = \text{softmax}(a^\top [W h_{v_i}^{(l-1)} \oplus W h_{v_j}^{(l-1)}] z_{v_i,v_j}^{(l)}) \tag{4}$$

Then, $\mathcal{L}_0$ can be differentiable over π. However, the downstream classification objective still does not allow for efficient gradient-based optimization. Therefore, we can further relax the binary gates z_{v_i,v_j}^l from Bernoulli distribution by a parameterized networks and employ hard-sigmoid rectification of the parameter $s_{v_i,v_j} \sim q_{s_{v_i,v_j}}(s_{v_i,v_j}|\phi)$ to mimic the binary gate $z_{v_i,v_j} = \min(1, \max(0, s_{v_i,v_j}))$ (Louizos et al., 2018; Maddison et al., 2017). Inspired by Louizos et al. (2018), we assume a binary concrete random variable s_{v_i,v_j} distributed in the (0,1) interval with probability density $q_{s_{v_i,v_j}}(s_{v_i,v_j}|\phi)$ and then utilize hard-sigmoid on s_{v_i,v_j}. The parameters of the distribution $\phi = (\tilde{\alpha}_{v_i,v_j}, \beta)$.

$$\begin{aligned} s_{v_i,v_j} &= \sigma((log u - log(1-u) + \tilde{\alpha}_{v_i,v_j})/\beta), \\ u &\sim \mathcal{U}(0,1),\ \bar{s} = s(\zeta - \gamma) + \gamma, \\ \tilde{\alpha}_{v_i,v_j} &= \text{leakyReLU}(a^\top [W \mathbf{h_{v_i}} \oplus W \mathbf{h_{v_j}}]), \\ z_{v_i,v_j} &= \min(1, \max(0, \bar{s}_{v_i,v_j})), \end{aligned} \tag{5}$$

where $\zeta < 0$ and $\gamma > 1$ are hyperparameters, and σ denotes sigmoid function. We stretch the distribution of s from the (0,1) interval to the (ζ, γ) interval. β is the temperature that controls the degree of approximation. If $\beta = 0$, we will get the bernoulli random variable $s_{v_i,v_j} = \pi_{v_i,v_j}$. And then, the sentence embeddings $\tilde{\mathbf{h}}_{\mathbf{v_i}}^{(\mathbf{l})}$ and output the new adjacency matrix $\tilde{A} = A \odot \mathbb{1}[Z > 0]$ can be rewritten as follows.

$$\tilde{\mathbf{h}}_{\mathbf{v_i}}^{(\mathbf{l})} = \sum_{v_j \in \mathcal{N}(v_i)} \text{softmax}(z_{v_i,v_j}) \mathbf{h}_{\mathbf{v_j}}^{(\mathbf{l-1})} \tag{6}$$

4.2.3 PREDICTION COMPONENT

The prediction component includes sentence-level and abstract-level prediction tasks, such as sentence identification, sentence stance prediction, abstract selection, and abstract stance prediction.

The new adjacency matrix $\tilde{A}$ and topology-consolidated evidence representation $\tilde{\mathbf{h}}_{\mathbf{v_i}}^{(\mathbf{l})}$ are fed into the GAT layer and feedforward layer. Abstract-level prediction component employs two feedforward

layers. The input of the abstract-level prediction component is the combination of the sentence-level output and the first token of the PLM hidden states, h_{cls}. Cross-entropy loss is used for abstract selection, stance prediction, and rationale selection. The final objective function is the weighted summation $\mathcal{L} = \lambda_1 \mathcal{L}_{abstract\ retrieval} + \lambda_2 \mathcal{L}_{stance} + \lambda_3 \mathcal{L}_{rationale\ selection} + \lambda_0 \mathcal{L}_0$.

5 EXPERIMENTS

5.1 DATASET

We use the SciFact [1] dataset, which contains 1,409 scientific claims related to COVID-19, verified by 5183 titles and abstracts from CORD-19 corpus, indexed from PubMed (Wadden et al., 2020). In reality, a claim can be both supported and refuted by different abstract rationales. However, in this dataset, all the selected rationale sentences for a claim have the same annotated label: "Supports", "NoInfo", or "Refutes". We only involve the training data and development data from the SciFact dataset, since we cannot evaluate the test data without ground truth labels. Accordingly, the training data and test data in total include 456 claims annotated as Supports, 416 claims labeled as NoInfo, and 237 claims labeled as Refutes, as shown in Table 1. Claims with the Supports or Refutes label contain at least one evidence abstract and a rationale sentence. We use 809 instances of training data to train our model and test it on the development data, which consists of 300 instances.

Table 1: Data Statistics of SciFact

Data	Supports	NoInfo	Refutes	All
Train	332	304	173	809
Dev	124	112	64	300
All	456	416	237	1109

5.2 BASELINE MODELS

We compare our framework with three baseline models, including KGAT and Paragraph-Joint (Li et al., 2021) and ARSJoint (Zhang et al., 2021).

- KGAT (Liu et al., 2020) employs a kernel mechanism to learn the node and edge masks during training. The training process of evidence abstract retrieval, rationale sentence selection, and stance prediction is in a pipeline fashion. That is, each task is trained separately. KGAT reranks the predicted evidence abstract and utilizes the top three evidence abstracts after reranking to train the sentence-level task.
- Paragraph-Joint (Li et al., 2021) is trained in a paragraph-level multi-task learning model.
- ARSJoint (Zhang et al., 2021) is the multi-task learning model for abstract retrieval, rationale identification, and stance prediction.

5.3 EVALUATION METRICS

We evaluate model performance at two levels of granularity: abstract and sentence levels, following the setup in (Wadden et al., 2020). For both levels, we report precision, recall, and micro-F1 scores. For abstract-level evaluation, we assess the prediction of the stance label only and of both the stance label and the rationale sentences. For sentence-level evaluation, we include only the evaluation of rationale sentence selection and the evaluation of both rationale sentence selection and stance prediction.

5.4 IMPLEMENTATION DETAILS

We first retrieve the top 150 related abstracts for each given claim in the training and testing datasets by using a bi-gram tf-idf vectorizer and calculating cosine similarity over the sum of title and abstract vectors.

[1] https://github.com/allenai/scifact

We train our model with $k = 12$ negative candidate abstracts for each given claim that are randomly selected from 150 related abstracts and 1 positive abstract sample. For the testing phase, we set k equal to 12, by randomly selecting 12 samples from 150 related abstracts for each claim. The epoch number of the training is 40. The learning rate to update the pre-trained language models is $1e-5$, and the learning rate of other parameters in our framework is $5e-6$. As for the pre-trained language models, we choose the Microsoft PubMedBert (Gu et al., 2021) as the backbone PLMs for PrunE and all baselines. PubMedBert is pretrained on PubMed titles, abstracts, and full texts. The parameter dimension of PubMedBert is 768. The hidden state dimension of PrunE is 512, with 1024 input dimension, following (Wadden et al., 2020; Li et al., 2021). We set the $\lambda_0 = 1$, $\lambda_1 = 3.5$, $\lambda_2 = 2.5$, $\lambda_3 = 4.5$ for the objective function. The topology adaptation component contains hyperparameter $\zeta = -1$, $\gamma = 1.1$, and temperature $\beta = 2/3$. The entire experiments are conducted on an Nvidia A100 GPU with 32GB of memory. The implementation scripts can be found at `https://github.com/LiriFang/PrunE-code.git`.

5.5 Experimental results

Overall Performance. PrunE achieves the competitive performance compared with baseline models in Table 2, and no baselines can outperform across all settings, which leaves more investigation space for future directions.

Table 2: Performance Comparison: The highest scores are in **bold**, second-highest are underlined.

Model	Sentence Level					
	Selection Only			Selection + Label		
	Precision	Recall	F1	Precision	Recall	F1
KGAT	0.6741	0.4126	0.5119	**0.5759**	0.3525	0.4373
Paragraph-Joint	**0.8084**	0.5765	**0.6730**	0.4713	0.3361	0.3923
ARSJoint	0.6380	0.5683	0.6012	0.4387	0.3907	0.4133
PrunE	0.6782	**0.5874**	0.6296	0.5741	**0.4973**	**0.5329**
Model	**Abstract Level**					
	Label Only			Label + Rationale		
	Precision	Recall	F1	Precision	Recall	F1
KGAT	**0.7227**	0.4115	0.5244	**0.6891**	0.3923	0.5000
Paragraph-Joint	0.5839	0.4163	0.4860	0.5369	0.3828	0.4469
ARSJoint	0.6012	0.4976	0.5445	0.5145	0.4258	0.4660
PrunE	0.6893	**0.5837**	**0.6321**	0.6554	**0.5550**	**0.6010**

Sentence-level evaluation. The F1 score of PrunE does not compete with the Paragraph-Joint in the rationale selection-only setting. However, PrunE achieves better precision, recall, and F1 scores than Paragraph-Joint and ARSJoint models, if both the rationale sentence and stance label are evaluated. Even though Paragraph-Joint achieves the highest performance in the selection-only setting, the recall of rationale sentences that have an accurate stance is lowest.

Abstract-level evaluation. PrunE achieves the best F1 score for abstract stance Label Only and Label + Rationale. Label + Rationale evaluates whether the abstract stance is correct and contains at least an accurate rationale sentence. KGAT utilizes rerank methods as an additional step for evidence abstract retrieval and uses only the top three evidence abstracts as training instances for rationale selection. The rerank module may contribute to the high precision score at an abstract level. Compared to the two joint training models, i.e., Paragraph-Joint and ARSJoint, PrunE achieves higher precision, recall, and F1 scores in both label-only and label+Rationale settings.

5.6 Ablation study

We conduct an ablation study to assess the effectiveness of the topology adaptation component in our framework. Specifically, we compare three variants: (1) the full framework without sparsification,

Table 3: Ablation Study: Performance of PrunE with various components removed.

Model	**Sentence Level**					
	Selection Only			Selection + Label		
	Precision	Recall	F1	Precision	Recall	F1
PrunE	**0.6782**	0.5874	**0.6296**	**0.5741**	**0.4973**	**0.5329**
PrunE w/o sparsification	0.4141	0.1448	0.2146	0.2969	0.1038	0.1538
PrunE (w/o topology adaptation)	0.6017	0.5902	0.5959	0.4318	0.4235	0.4276
PrunE (w/o complete graphs)	0.6083	**0.5984**	0.6033	0.4444	0.4372	0.4408
Model	**Abstract Level**					
	Label Only			Label + Rationale		
	Precision	Recall	F1	Precision	Recall	F1
PrunE	**0.6893**	**0.5837**	**0.6321**	**0.6554**	**0.5550**	**0.6010**
PrunE w/o sparsification	0.6154	0.1148	0.1935	0.5128	0.0957	0.1613
PrunE (w/o topology adaptation)	0.5926	0.5359	0.5628	0.5450	0.4928	0.5176
PrunE (w/o complete graphs)	0.5735	0.5598	0.5666	0.5196	0.5072	0.5072

(2) the framework without the entire topology adaptation module (as illustrated in Figure 2), and (3) the framework without constructing the complete evidence graph.As shown in Table 3, the model variant that stacks three GAT layers over a fully connected graph, without learning a sparse adjacency matrix, performs significantly worse than the full model. This might be caused by stacking three GAT layers over a complete graph with small sizes (around 10 nodes), leading to an over-smoothing problem for sentence-level prediction tasks. Moreover, our abstract-level embeddings are generated based on sentence selection rationale, which can propagate the error to the abstract prediction tasks. The performance increases when we drop the entire topology adaptation component; however, it is still lower than PrunE. Lastly, removing the complete evidence graph also results in a performance drop, comparable to the model without the topology adaptation component. These results collectively demonstrate that both the construction of the evidence graph and the topology adaptation mechanism make meaningful contributions to model performance.

6 Conclusion and Discussion

We propose PrunE, a lightweight and structure-aware framework for scientific claim verification that integrates graph representation learning with pretrained language models. PrunE comprises three main components: (1) sentence embeddings initialized using PLMs and attention pooling; (2) a fully connected evidence graph that is pruned via learnable binary gates to capture task-relevant structural relationships; and (3) a prediction module that leverages the refined embeddings and the learned graph structure to jointly identify relevant abstracts, rationale sentences, and their stance toward a given claim.

Our method addresses two central challenges in scientific claim verification: the complex, non-sequential structure of scientific text, and the need to augment general-domain PLMs with domain-specific relational knowledge without incurring the high cost of fine-tuning or large-scale retrieval. By learning sparse and interpretable graph structures, PrunE enables localized multi-hop reasoning over sentence-level evidence, while remaining compatible with lightweight, modular inference pipelines.

This work aims to contribute to the growing body of research at the intersection of graph learning and language modeling, highlighting opportunities for enhancing retrieval-augmented and instruction-tuned LLMs with structural priors. Future directions include modeling discourse-aware graph topologies and integrating symbolic or citation-based constraints to further guide the construction of evidence graphs.

REFERENCES

Shashank Agarwal and Hong Yu. Automatically classifying sentences in full-text biomedical articles into introduction, methods, results and discussion. *Bioinform.*, 25(23):3174–3180, 2009. doi: 10.1093/bioinformatics/btp548. URL https://doi.org/10.1093/bioinformatics/btp548.

Carlos Alvarez, Maxwell Bennett, and Lucy Wang. Zero-shot scientific claim verification using LLMs and citation text. In *Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024)*, pp. 269–276, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.sdp-1.25/.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015. URL http://arxiv.org/abs/1409.0473.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423. URL https://doi.org/10.18653/v1/n19-1423.

Jay DeYoung, Eric Lehman, Benjamin E. Nye, Iain James Marshall, and Byron C. Wallace. Evidence inference 2.0: More data, better models. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing, BioNLP 2020*, pp. 123–132, 2020. URL https://doi.org/10.18653/v1/2020.bionlp-1.13.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pp. 1107–1128. Association for Computational Linguistics, 2024. URL https://aclanthology.org/2024.emnlp-main.64.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2024.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 6491–6501, 2024.

Dongqi Fu and Jingrui He. SDG: A simplified and dynamic graph neural network. In *SIGIR*, 2021.

Dongqi Fu, Yikun Ban, Hanghang Tong, Ross Maciejewski, and Jingrui He. DISCO: comprehensive and explainable disinformation detection. In *CIKM*, 2022a.

Dongqi Fu, Liri Fang, Ross Maciejewski, Vetle I. Torvik, and Jingrui He. Meta-learned metrics over multi-evolution temporal graphs. In *KDD*, 2022b.

Dongqi Fu, Liri Fang, Zihao Li, Hanghang Tong, Vetle I. Torvik, and Jingrui He. What do llms need to understand graphs: A survey of parametric representation of graphs. *CoRR*, 2024a.

Dongqi Fu, Zhigang Hua, Yan Xie, Jin Fang, Si Zhang, Kaan Sancak, Hao Wu, Andrey Malevich, Jingrui He, and Bo Long. Vcr-graphormer: A mini-batch graph transformer via virtual connections. In *ICLR*, 2024b.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2, 2023.

Max Glockner, Ivan Habernal, and Iryna Gurevych. Why do you think that? exploring faithful sentence-level rationales without supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020*, 2020. URL https://doi.org/10.18653/v1/2020.findings-emnlp.97.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1), 2021. doi: 10.1145/3458754. URL https://doi.org/10.1145/3458754.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *CoRR*, abs/2403.14608, 2024. doi: 10.48550/ARXIV.2403.14608. URL https://doi.org/10.48550/arXiv.2403.14608.

Hyewon Jeon and Jay-Yoon Lee. Graphcheck: Multi-path fact-checking with entity-relationship graphs. *CoRR*, abs/2502.20785, 2025. doi: 10.48550/ARXIV.2502.20785. URL https://doi.org/10.48550/arXiv.2502.20785.

Neema Kotonya and Francesca Toni. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7740–7754, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.623. URL https://aclanthology.org/2020.emnlp-main.623.

Mufei Li, Siqi Miao, and Pan Li. Simple is effective: The roles of graphs and large language models in knowledge-graph-based retrieval-augmented generation. *arXiv preprint arXiv:2410.20724*, 2024a.

Xiangci Li, Gully A. Burns, and Nanyun Peng. A paragraph-level multi-task learning model for scientific fact-verification. In *Proceedings of the Workshop on Scientific Document Understanding co-located with 35th AAAI Conference on Artificial Inteligence, SDU@AAAI 2021, Virtual Event, February 9, 2021*, volume 2831 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2021.

Zihao Li, Lecheng Zheng, Bowen Jin, Dongqi Fu, Baoyu Jing, Yikun Ban, Jingrui He, and Jiawei Han. Can graph neural networks learn language with extremely weak text supervision? *arXiv preprint arXiv:2412.08174*, 2024b.

Zihao Li, Dongqi Fu, Mengting Ai, and Jingrui He. Apex^2: Adaptive and extreme summarization for personalized knowledge graphs. In *KDD*, 2025.

Jiawei Liu, Cheng Yang, Zhiyuan Lu, Junze Chen, Yibo Li, Mengmei Zhang, Ting Bai, Yuan Fang, Lichao Sun, Philip S Yu, et al. Graph foundation models: Concepts, opportunities and challenges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. Fine-grained fact verification with kernel graph attention network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pp. 7342–7351, 2020. URL https://doi.org/10.18653/v1/2020.acl-main.655.

Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l_0 regularization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. URL https://openreview.net/forum?id=H1Y8hhg0b.

Dongsheng Luo, Wei Cheng, Wenchao Yu, Bo Zong, Jingchao Ni, Haifeng Chen, and Xiang Zhang. Learning to drop: Robust graph neural network via topological denoising. In *WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021*, pp. 779–787, 2021. doi: 10.1145/3437963.3441734. URL https://doi.org/10.1145/3437963.3441734.

Jing Ma, Wei Gao, and Kam-Fai Wong. Detect rumor and stance jointly by neural multi-task learning. In *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pp. 585–593, 2018. URL https://doi.org/10.1145/3184558.3188729.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017. URL https://openreview.net/forum?id=S1jE5L5gl.

William C Mann and Sandra A Thompson. Rhetorical structure theory: A framework for the analysis of texts. Technical report, UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY INFORMATION SCIENCES INST, 1987.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications. *CoRR*, abs/2402.07927, 2024. doi: 10.48550/ARXIV.2402.07927. URL https://doi.org/10.48550/arXiv.2402.07927.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019*, 2019. URL https://doi.org/10.1145/3292500.3330935.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, Volume 1 (Long Papers)*, 2018.

Katherine Tieu, Dongqi Fu, Yada Zhu, Hendrik F. Hamann, and Jingrui He. Temporal graph neural tangent kernel with graphon-guaranteed. In *NeurIPS*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Juraj Vladika, Ivana Hacajová, and Florian Matthes. Step-by-step fact verification system for medical claims with explainable reasoning. *CoRR*, abs/2502.14765, 2025. doi: 10.48550/ARXIV.2502.14765. URL https://doi.org/10.48550/arXiv.2502.14765.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, 2020. URL https://doi.org/10.18653/v1/2020.emnlp-main.609.

David Wadden, Kyle Lo, Lucy Lu Wang, Arman Cohan, Iz Beltagy, and Hannaneh Hajishirzi. MultiVerS: Improving scientific claim verification with weak supervision and full-document

context. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 61–76, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.6. URL https://aclanthology.org/2022.findings-naacl.6.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. Instruction tuning for large language models: A survey. *CoRR*, abs/2308.10792, 2023. doi: 10.48550/ARXIV.2308.10792. URL https://doi.org/10.48550/arXiv.2308.10792.

Zhiwei Zhang, Jiyi Li, Fumiyo Fukumoto, and Yanming Ye. Abstract, rationale, stance: A joint model for scientific claim verification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 3580–3586, 2021. doi: 10.18653/v1/2021.emnlp-main.290. URL https://doi.org/10.18653/v1/2021.emnlp-main.290.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.

Lecheng Zheng, Dongqi Fu, Ross Maciejewski, and Jingrui He. Drgnn: Deep residual graph neural network with contrastive learning. *Transactions on Machine Learning Research*, 2024.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. Reasoning over semantic-level graph for fact checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pp. 6170–6180, 2020. URL https://doi.org/10.18653/v1/2020.acl-main.549.

Dawei Zhou, Lecheng Zheng, Dongqi Fu, Jiawei Han, and Jingrui He. Mentorgnn: Deriving curriculum for pre-training gnns. In *CIKM*, 2022.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. GEAR: graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Volume 1: Long Papers*, pp. 892–901, 2019. URL https://doi.org/10.18653/v1/p19-1085.

Xi Zhu, Haochen Xue, Ziwei Zhao, Wujiang Xu, Jingyuan Huang, Minghao Guo, Qifan Wang, Kaixiong Zhou, and Yongfeng Zhang. Llm as gnn: Graph vocabulary learning for text-attributed graph foundation models. *arXiv preprint arXiv:2503.03313*, 2025.

Jiaru Zou, Dongqi Fu, Sirui Chen, Xinrui He, Zihao Li, Yada Zhu, Jiawei Han, and Jingrui He. Gtr: Graph-table-rag for cross-table question answering. *arXiv preprint arXiv:2504.01346*, 2025.