Subgraph Retrieval Enhanced by Graph-Text Alignment for Commonsense Question Answering

Boci Peng[1,2], Yongchao Liu[2], Xiaohe Bo[3], Tian Sheng[2], Baokun Wang[2], Chuntao Hong[2], Yan Zhang[1]
1 School of Intelligence Science and Technology, Peking University
2 Ant Group
3 School of Artificial Intelligence, Beijing Normal University

Background—Commonsense Question Answering (CSQA)

- **CSQA** is a crucial task in natural language understanding that requires reasoning according to commonsense knowledge

- Existing CSQA datasets generally adopt **multiple-choice questions** to evaluate the model's performance

Where is the capital of China?

A. London.

B. Beijing.

C. New York

D. Shanghai.

E. Guangzhou.

Background—Commonsense Question Answering (CSQA)

- **Challenge**: It is difficult to learn commonsense knowledge solely from pre-training text corpora, as it is rarely expressed explicitly in natural language
- **Knowledge Graph**: Knowledge graphs are more efficient in representing commonsense and can aid PLMs in comprehending QA pairs and enhancing reasoning capabilities
- **Extracting-and-Modeling Paradigm**: Existing KG-augmented works primarily follow a paradigm that first extracts relevant subgraphs or paths related to a given question based on pre-defined rules, and then models the extracted structural knowledge

Background—Limitations of Previous Methods

- **Subgraph Quality**: The subgraph's quality suffers when retrieved through simple string or semantic matching, posing limitations for subsequent operations

- **Graph-Text Misalignment**: The misalignment between graph and text encoders presents a challenge for PLMs to internalize the knowledge contained in the acquired subgraph, leading to reduced task performance

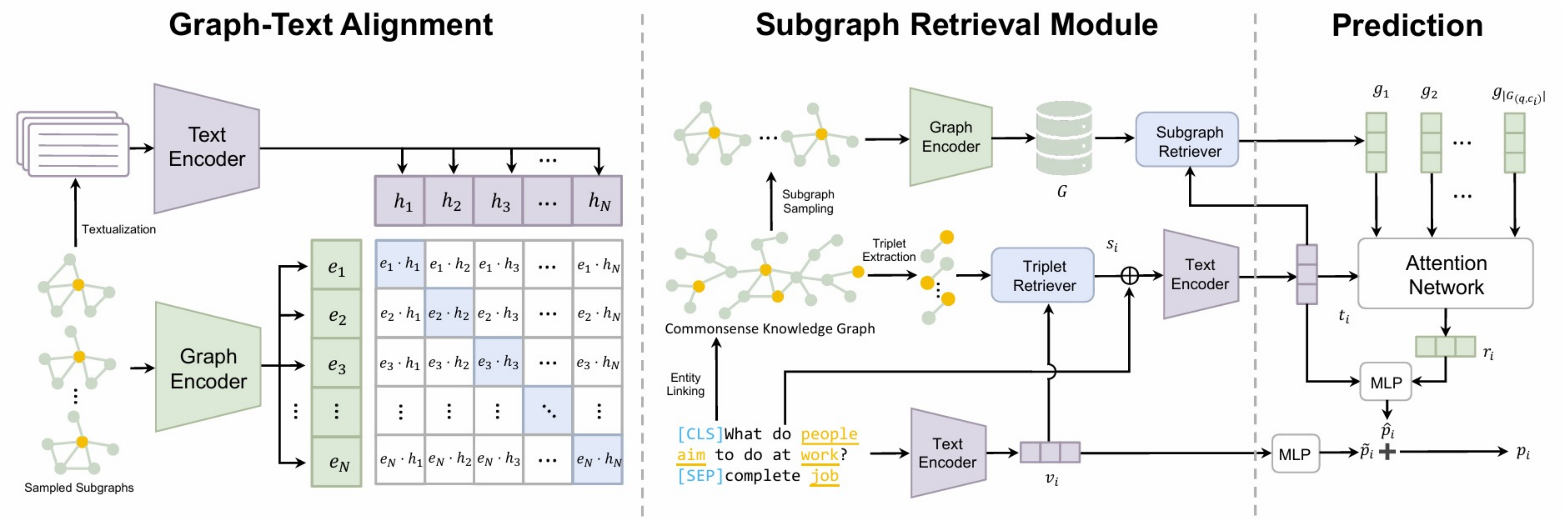
- **Uncontrolled Subgraph Size**: To obtain sufficient relevant knowledge, the number of nodes in the subgraph will expand dramatically with the increase of hop count, raising the burden of the model

Motivation

- **Subgraph Vector Database:** To address the limitations of rule-based subgraph extraction methods that may overlook critical nodes and result in uncontrollable subgraph size
- **BFS-style Subgraph Sampling:** To ensure complete neighbor information for each node and avoid the blockage of the message-passing mechanism of GNNs caused by pruning edges linked to marginal nodes
- **Bidirectional Contrastive Learning:** To overcome the challenge of misalignment between graph and text encoders, which undermines the effectiveness of knowledge fusion and impacts task performance

蚂蚁集团 ANT GROUP | TuGraph™

Model Architecture



➢ A bidirectional contrastive method is proposed to align the semantic space of graph and text encoders

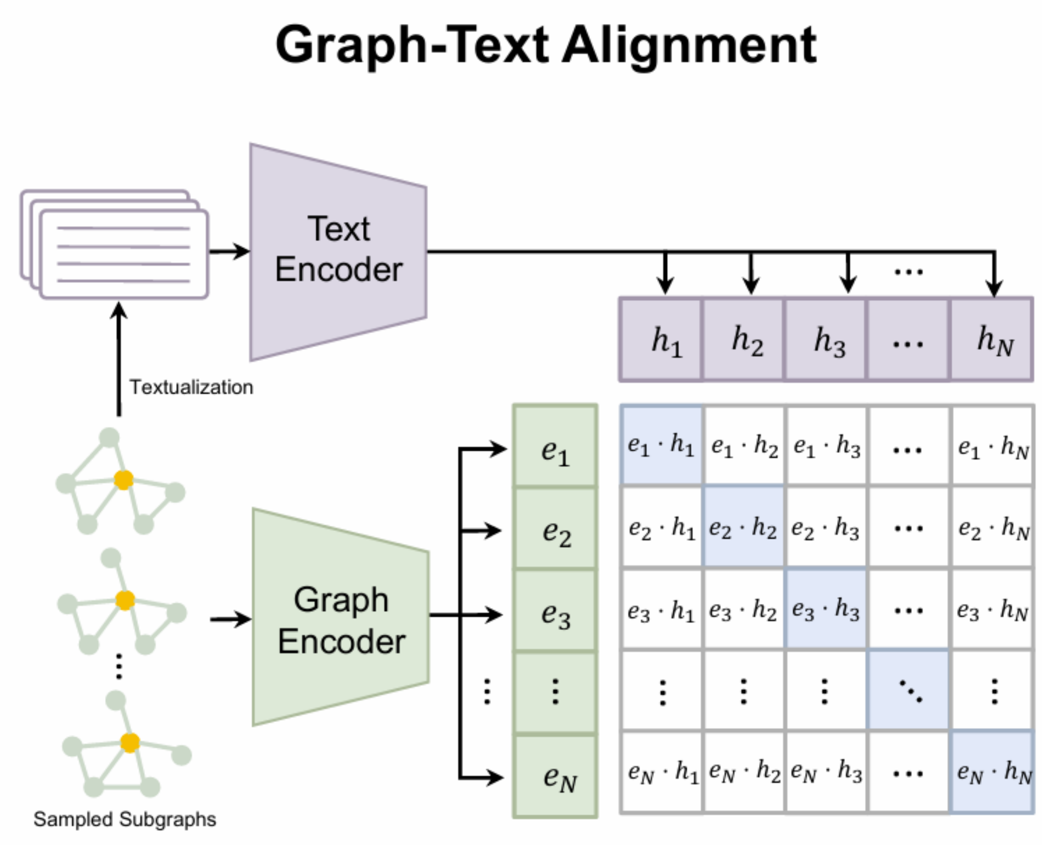
➢ Transform the knowledge graph into a subgraph vector database

➢ Introduce a query enhancement strategy for better subgraph retrieval

➢ All the information retrieved is combined by an attention mechanism to bolster the reasoning ability of PLMs

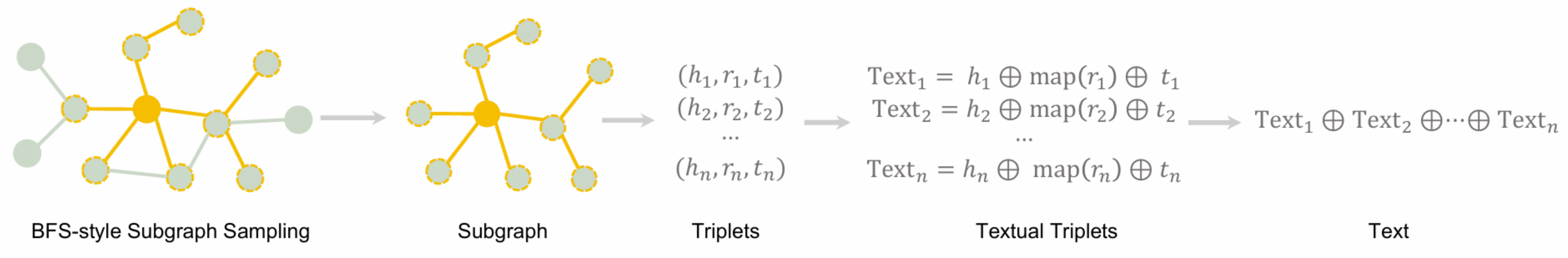
Graph-Text Alignment

- **Motivation**: Coordinate the embedding spaces of graph and text encoders and fully harness the respective strengths of text and KG
- **Method**:
  - Generate training graph-text pairs with equivalent semantics
  - Employ a bidirectional contrastive learning method to train the encoders of both modalities



Construction of Graph-Text Pairs

- A BFS-style sampling strategy for subgraph construction, which initiates from the central node and proceeds to sample neighbors hop-by-hop
- Textualize the subgraphs to construct synonymous text descriptions
  - Convert all relation links into triplet descriptions: Map each relation type to a relation template and concatenate the head concept, relation template, and tail concept as the description of each triplet
  - Concatenate all descriptions to compose the final description



Graph-Text Contrastive Learning

- GNN and PLM are utilized to encode the knowledge subgraphs and natural language descriptions to obtain the corresponding representation

$$\tilde{\boldsymbol{e}}_{\boldsymbol{i}} = \mathrm{Pool}_G(\mathrm{GNN}(\mathcal{G}_i)),$$
$$\tilde{\boldsymbol{h}}_{\boldsymbol{i}} = \mathrm{Pool}_T(\mathrm{PLM}(s_i)),$$

- To project the knowledge subgraph embedding and text embedding into the same semantic space, two linear projection layers are designed as follows:

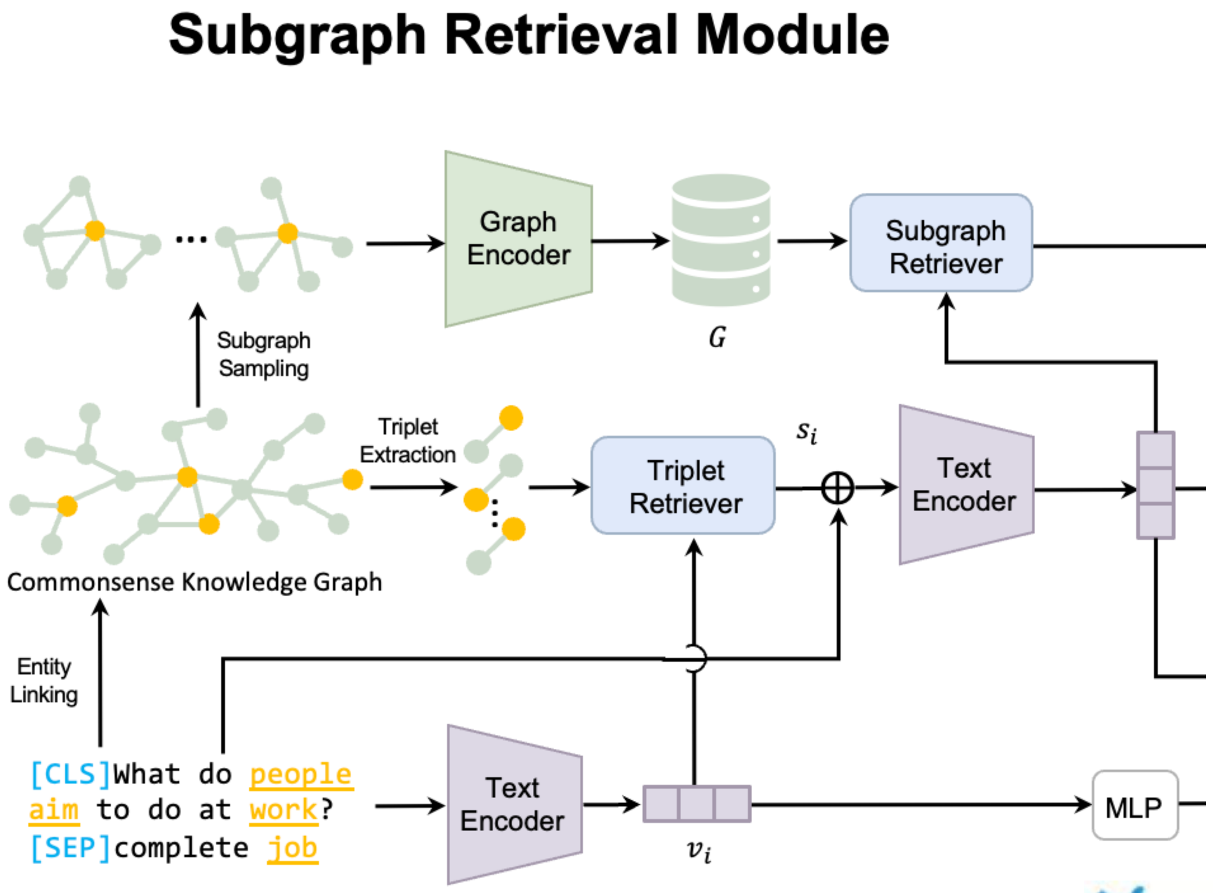
$$\boldsymbol{e}_i = \boldsymbol{W}_G \tilde{\boldsymbol{e}}_{\boldsymbol{i}} + \boldsymbol{b}_G,$$
$$\boldsymbol{h}_i = \boldsymbol{W}_T \tilde{\boldsymbol{h}}_{\boldsymbol{i}} + \boldsymbol{b}_T,$$

- Employ InfoNCE with in-batch negative sampling to align representations of two modalities bidirectionally

$$\mathcal{L}_{\mathrm{G2T}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(sim(\boldsymbol{e}_i, \boldsymbol{h}_i)/\tau)}{\sum_{j=1}^{N}\exp(sim(\boldsymbol{e}_i, \boldsymbol{h}_j)/\tau)}$$
$$\mathcal{L}_{\mathrm{T2G}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(sim(\boldsymbol{h}_i, \boldsymbol{e}_i)/\tau)}{\sum_{j=1}^{N}\exp(sim(\boldsymbol{h}_i, \boldsymbol{e}_j)/\tau)}$$
$$\mathcal{L}_{GT} = \frac{1}{2}(\mathcal{L}_{G2T} + \mathcal{L}_{T2G})$$

Subgraph Retrieval Module

- Subgraph vector database construction
- Query enhancement
- Subgraph retrieval



Database Construction

- **BFS-style Sampling**: We adopt a BFS-style subgraph sampling strategy which is the same as the graph-text pairs construction, leveraging the analogy between BFS and the message-passing mechanism of GNNs

- **Subgraph Vector**: For each subgraph, we obtain its graph embedding $\boldsymbol{e}_i$ and text embedding $\boldsymbol{h}_i$, and combine them to form the subgraph vector:

$$\boldsymbol{g}_i = \frac{1}{2}\left(\frac{\|\boldsymbol{h}_i\|}{\|\boldsymbol{e}_i\|}\boldsymbol{e}_i + \boldsymbol{h}_i\right)$$

- **Vector Database**: We construct a subgraph vector database $\boldsymbol{G} = \{\boldsymbol{g}_i\}_{i=1}^{|G|}$ with all subgraph vectors

蚂蚁集团 ANT GROUP | TuGraph™

Query Enhancement

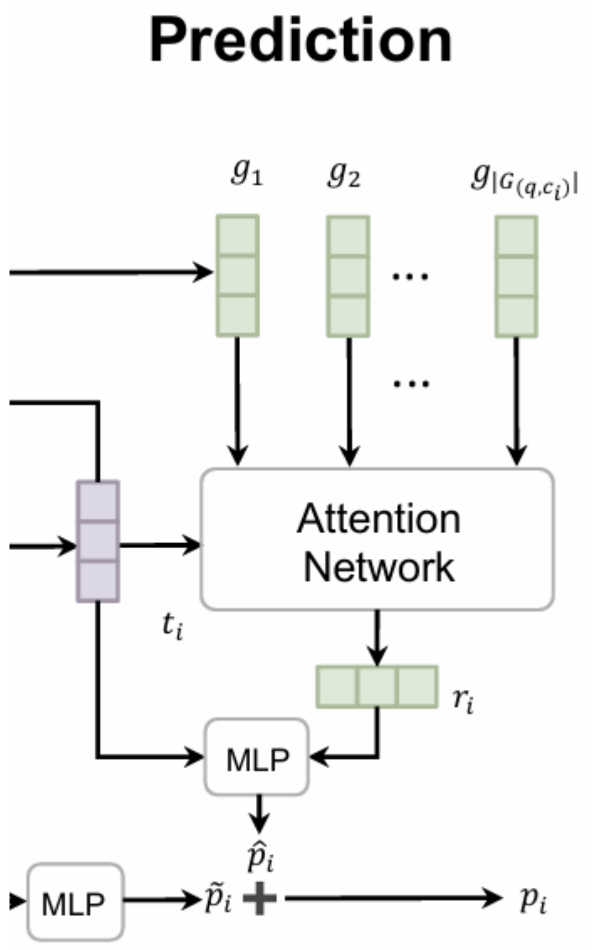
- **Challenge**: Direct use of Q-A pair embeddings as queries may not align well with the pre-trained corpus, affecting retrieval accuracy
- **Enhancement:** Retrieve question-related triplets from the KG and concatenate them with Q-A pairs
- **Entity Linking:** Apply entity linking to find entities in the question and options, and retrieve triplets containing these entities
- **Concatenation**: Concatenate the retrieved fact triplets with the question and options, termed as s_i
- **Encoding**: Use the aligned PLM to encode s_i into $\boldsymbol{t}_i$, which serves as the enhanced query for subgraph retrieval

Subgraph Retrieval

- **Retrieval:** With the enhanced query $\boldsymbol{t}_i$, we retrieve relevant subgraph vectors from the subgraph vector database $\boldsymbol{G}$ based on cosine similarity
- **Top-$\boldsymbol{k}$**: We recall the top-k subgraph vectors with the highest similarities, denoted as $\boldsymbol{G}_{q,c_i}$

Prediction

- **Integration**: Integrate the retrieved subgraph vectors through multi-head attention with $\boldsymbol{t}_i$ as the query
- **Score Prediction**: Add the integrated representation and the enhanced query, and feed them into a linear layer to predict the score of the option
- **Direct Inference**: Since some questions are expected to be answered based solely on the question context, we also encode the Q-A pair directly to infer the score without additional knowledge
- **Final Score**: The two scores are weighted and summed to yield the final score



Experimental Setup

- **Datasets**
  - ✓ CommonsenseQA: 5-way multiple-choice QA dataset, including the official split and the in-house split
  - ✓ OpenBookQA: a 4-choice dataset to evaluate the science commonsense knowledge
  - ✓ SocialIQA: a 3-choice dataset to evaluate the understanding of commonsense social knowledge
  - ✓ PIQA: a 2-choice QA dataset regarding physical commonsense
  - ✓ RiddleSenseQA: a 5-choice QA dataset about commonsense riddles

Task	Train	Dev	Test
CommonsenseQA official split	9,741	1,221	1,140
CommonsenseQA in-house split	8,500	1,221	1,241
OpenBookQA	4,957	500	500
SocialIQA	33,410	1,954	-
PIQA	16,113	1,838	-
RiddleSenseQA	3,510	1,021	-

- **Metrics**: Accuracy

Comparison with baselines

Methods	CommonsenseQA		OpenBookQA	
	IHdev-Acc (%)	IHtest-Acc (%)	RoBERTa-Large (%)	AristoRoBERTa (%)
Fine-tuned LMs	73.07 (±0.45)	68.69 (±0.56)	64.80 (±2.37)	78.40 (±1.64)
+ RN	74.57 (±0.91)	69.08 (±0.21)	65.20 (±1.18)	75.35 (±1.39)
+ RGCN	72.69 (±0.19)	68.41 (±0.66)	62.45 (±1.57)	74.60 (±2.53)
+ GconAttn	72.61 (±0.39)	68.59 (±0.96)	64.75 (±1.48)	71.80 (±1.21)
+ MHGRN	74.45 (±0.10)	71.11 (±0.81)	66.85 (±1.19)	80.60
+ QA-GNN	76.54 (±0.21)	73.41 (±0.92)	67.80 (±2.75)	82.77 (±1.56)
+ DGRN	78.20	74.00	69.60	84.10
+ GreaseLM	78.50 (±0.50)	74.20 (±0.40)	68.80 (±1.75)	84.80
+ JointLK	77.88 (±0.25)	74.43 (±0.83)	70.34 (±0.75)	84.92 (±1.07)
+ GSC	79.11 (±0.22)	74.48 (±0.41)	70.33 (±0.81)	86.67 (±0.46)
+ SAFE	76.93 (±0.37)	74.03 (±0.43)	69.20	87.13
+ HamQA	76.88	73.91	71.12	84.59
+ DRAGON*	-	**76.00**	72.00	-
+ DRAGON (w/o MLM)*	-	73.80	66.40	-
+ DHLK*	79.39 (±0.24)	74.68 (±0.26)	72.20 (±0.40)	86.00 (±0.79)
+ SEPTA (**Ours**)	**79.61** (±0.17)	74.78 (±0.23)	**72.33** (±0.35)	**87.37** (±0.51)

- Our method can contribute performance gains to LMs
- SEPTA outperforms all baselines without additional corpus on both datasets
- Compared to baselines incorporating additional corpus, our method also achieves comparable performance

Leaderboard

Methods	Test-Acc (%)
RoBERTa [17]	72.1
RoBERTa+FreeLB	72.2
RoBERTa+HyKAS	73.2
RoBERTa+KE	73.3
RoBERTa+KEDGN	74.4
RoBERTa+MHGRN [9]	75.4
RoBERTa+QA-GNN [34]	76.1
RoBERTa+GSC [29]	76.2
Albert	73.5
ALBERT+Path Generator [30]	75.6
ALBERT+HGN [9]	77.3
UnifiedQA (11B) [14]	**79.1**
RoBERTa+SEPTA (**Ours**)	76.6

Methods	Test-Acc (%)
Careful Selection [1]	72.0
AristoRoBERTa [6]	77.8
KF+SIR	80.0
AristoRoBERTa+PG [30]	80.2
AristoRoBERTa+MHGRN [9]	80.6
AristoRoBERTa+QA-GNN [34]	82.8
AristoRoBERTa+GreaseLM [36]	84.8
AristoRoBERTa+GSC [29]	87.4
AristoRoBERTa+MVP-Tuning [11]	87.6
ALBERT + KB	81.0
T5	83.2
UnifiedQA (11B) [14]	87.2
AristoRoBERTa+SEPTA (**Ours**)	**87.8**

- Evaluate SEPTA on the official CommonsenseQA and OpenBookQA leaderboards
- Our method achieves results surpassing all baselines based on the identical PLM
- Exhibit comparative performance compared with methods with larger-scale parameters (e.g., UnifiedQA)

Other Datasets

Methods	SocialIQA	PIQA	RiddleSenseQA
RoBERTa-Large	78.25	77.53	60.72
+ GconAttn	78.86	78.24	61.77
+ RN	78.45	76.88	62.17
+ MHGRN	78.11	77.15	63.27
+ QA-GNN	78.10	78.24	63.39
+ GreaseLM	77.89	78.02	63.88
+ GSC	78.61	78.40	64.07
+ SAFE	78.86	79.43	63.78
+ SEPTA (**Ours**)	**79.21**	**80.85**	**67.62**

- SEPTA consistently achieves superior performance
- This observation underscores the overall effectiveness of SEPTA in addressing various commonsense reasoning datasets or tasks, demonstrating a unified methodology

Ablation Study

Ablation	CommonsenseQA	OpenBookQA
SEPTA	74.78	72.33
w/o alignment	69.83 (-4.95)	67.20 (-5.13)
w/o subgraph	72.34 (-2.44)	70.23 (-2.10)
w/o triplets	71.25 (-3.53)	69.67 (-2.66)
$\lambda = 1.0$	74.13 (-0.65)	70.47 (-1.86)

- Four components are all crucial for SEPTA, and removing any part will result in a decrease in performance
- The performance drops the most significantly when we remove the graph-text alignment
- Removing either fact triplets or subgraph vectors will affect the performance
- Only using knowledge-enhanced representations for predictions (i.e. λ=1.0) cannot achieve optimal results

Low-Resource Setting

Methods	CommonsenseQA						OpenBookQA					
	5%	10%	20%	50%	80%	100%	5%	10%	20%	50%	80%	100%
RoBERTa-large	29.66	42.84	58.47	66.13	68.47	68.69	37.00	39.4	41.47	53.07	57.93	64.8
+ RGCN	24.41	43.75	59.44	66.07	68.33	68.41	38.67	37.53	43.67	56.33	63.73	62.45
+ GconAttn	21.92	49.83	60.09	66.93	69.14	68.59	38.60	36.13	43.93	50.87	57.87	64.75
+ RN	23.77	34.09	59.90	65.62	67.37	69.08	33.73	35.93	41.40	49.47	59.00	65.20
+ MHGRN	29.01	32.02	50.23	68.09	70.83	71.11	38.00	36.47	39.73	55.73	55.00	66.85
+ QA-GNN	32.95	37.77	50.15	69.33	70.99	73.41	33.53	35.07	42.40	54.53	52.47	67.80
+ GreaseLM	22.80	56.16	63.09	70.56	73.41	74.20	39.00	39.60	42.20	57.87	65.13	68.80
+ GSC	31.02	35.07	65.83	70.94	73.82	74.48	29.60	41.80	42.40	58.03	65.97	70.33
+ SAFE	36.45	56.51	65.16	70.72	73.22	74.03	38.80	41.20	44.93	58.33	65.60	69.20
+ SEPTA(**Ours**)	**50.69**	**62.37**	**68.09**	**71.80**	**74.05**	**74.78**	**45.63**	**54.80**	**58.10**	**66.57**	**68.30**	**72.33**

- SEPTA achieves promising performance in all settings
- It exhibits a trend where the performance improvement relative to other baselines is more significant with fewer training

Conclusion

- We propose a novel framework called Subgraph REtrieval Enhanced by GraPh-Text Alignment (SEPTA) for commonsense question answering (CSQA)
- SEPTA reframes the CSQA task as a subgraph vector retrieval problem and introduces a graph-text alignment method to enhance retrieval accuracy and facilitate knowledge fusion for prediction
- Extensive experiments on five CSQA datasets demonstrate the effectiveness and robustness of the SEPTA framework, outperforming SOTA approaches

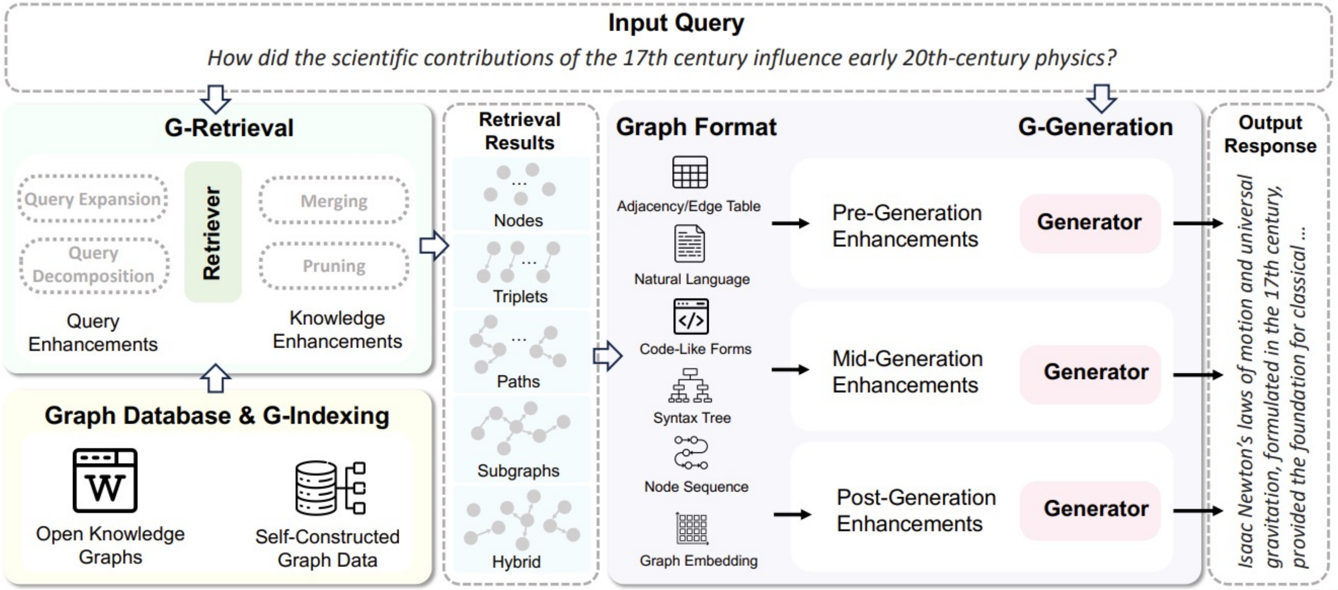
Future Works

- **Pre-training Tasks**: Explore more effective pre-training tasks for semantic alignment between graph and text representations
- **Larger Language Models**: Apply the SEPTA framework to larger language models if sufficient computational resources are available
- **Other Tasks**: Extend the SEPTA framework to other related tasks, such as node classification and link prediction on text-attributed graphs

GraphRAG Survey

Graph Retrieval-Augmented Generation: A Survey

BOCI PENG[*], School of Intelligence Science and Technology, Peking University, China
YUN ZHU[*], College of Computer Science and Technology, Zhejiang University, China
YONGCHAO LIU, Ant Group, China
XIAOHE BO, Gaoling School of Artificial Intelligence, Renmin University of China, China
HAIZHOU SHI, Rutgers University, US
CHUNTAO HONG, Ant Group, China
YAN ZHANG[†], School of Intelligence Science and Technology, Peking University, China
SILIANG TANG, College of Computer Science and Technology, Zhejiang University, China



url: https://arxiv.org/abs/2408.08921

Thanks for your listening!