CORN: Co-Reasoning Network for Commonsense Question Answering

Xin Guan, Biwei Cao, Qingqing Gao, Zheng Yin, Bo Liu, Jiuxin Cao*
Southeast University
{xin_guan,caobiwei,qingqing_gao,z.yin,bliu,jx.cao}@seu.edu.cn

Abstract

Commonsense question answering (QA) requires machines to utilize the QA content and external commonsense knowledge graph (KG) for reasoning when answering questions. Existing work uses two independent modules to model the QA contextual text representation and relationships between QA entities in KG, which prevents information sharing between modules for co-reasoning. In this paper, we propose a novel model, **Co**-**R**easoning **N**etwork (CORN), which adopts a bidirectional multi-level connection structure based on Co-Attention Transformer. The structure builds *bridges* to connect each layer of the text encoder and graph encoder, which can introduce the QA entity relationship from KG to the text encoder and bring contextual text information to the graph encoder, so that these features can be deeply interactively fused to form comprehensive text and graph node representations. Meanwhile, we propose a QA-aware node based KG subgraph construction method. The QA-aware nodes aggregate the question entity nodes and the answer entity nodes, and further guide the expansion and construction process of the subgraph to enhance the connectivity and reduce the introduction of noise. We evaluate our model on QA benchmarks in the CommonsenseQA and OpenBookQA datasets, and CORN achieves state-of-the-art performance.

1 Introduction

Commonsense Question answering (QA) research requires the machine to have a human thought pattern, which is capable of comprehending text content and combining commonsense knowledge to reason and arrive at the correct answer. Despite the success of large pre-trained language models (PLMs) (Devlin et al., 2019; Liu et al., 2019) on various NLP tasks, there is still a large gap between PLMs and humans on commonsense QA tasks. Therefore, researchers try to introduce external knowledge, such as Freebase (Bollacker et al., 2008) and ConceptNet (Speer et al., 2017), which are large knowledge graphs (KGs) that entities link by various relationships.

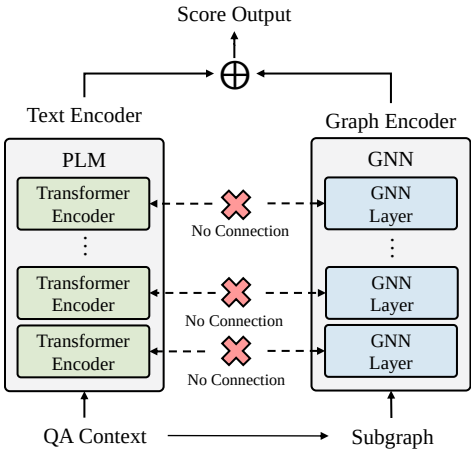

Figure 1: The network architecture of current commonsense QA research. Text encoder for encoding the QA content, graph encoder for reasoning on the graph. There is no connection between stacked text layers and stacked GNN layers.

There has already been a significant amount of work that combines PLMs and KGs for reasoning (Lin et al., 2019; Wang et al., 2020; Feng et al., 2020; Yasunaga et al., 2021). As illustrated in Figure 1, these works are mainly composed of two modules: (1) capturing text features on QA with a text encoder (such as PLM). (2) extracting subgraph from KG and reasoning on it with a graph encoder, such as the GNN-based model (Kipf and Welling, 2017; Schlichtkrull et al., 2018). Most work (Lin et al., 2019; Wang et al., 2020; Feng et al., 2020) focuses on building more efficient graph encoders to capture relationships between entities in graphs for reasoning. However, it ignores the interconnections between the QA content and graph due to GNN and PLM being treated as independent modules. To address the above problems,

*Corresponding author.

Proceedings of the 29th International Conference on Computational Linguistics, pages 1677–1686
October 12–17, 2022.

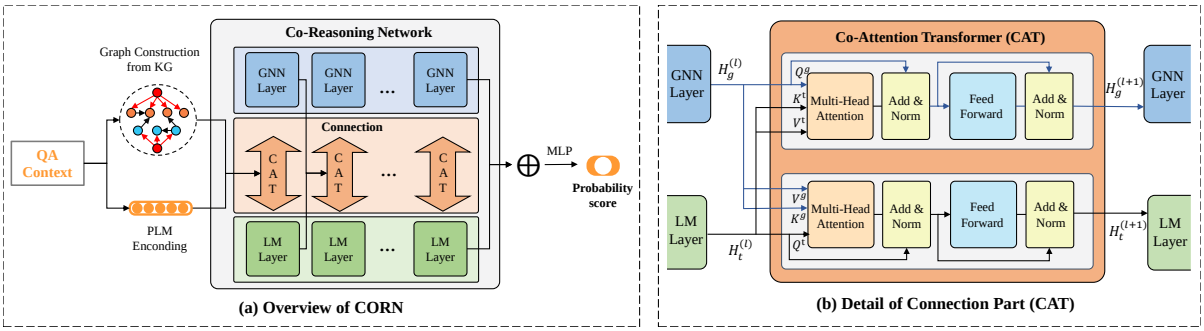

Figure 2: Overview of our model. **(a)** is the overall architecture of CORN. We first construct a subgraph related to QA by KG (§3.1), use PLM to encode the QA content, then apply N-layer Co-Reasoning Network, which combines GNN layer (§3.4) and LM layer (§3.3) through Co-Attention Transformer (CAT) (§3.2), to perform reasoning. **(b)** is the detail of connection part (Co-Attention Transformer).

Yasunaga et al. (2021) perform joint reasoning by explicitly adding the QA content to the graph in the form of a node. Nevertheless, this method is a one-way connection structure, only enabling the graph encoder to obtain textual context information for reasoning while the text encoder cannot perceive graph information.

We propose a novel model, **Co**-**R**easoning **N**etwork (CORN), to solve the above problems. CORN adopts a bidirectional multi-level connection structure, which connects the language model (LM) and GNN. Specifically, we build *bridge*s between each layer of these two types of models by the Co-Attention Transformer. Through this *bridge*, the text representation and graph node representation can be fused bidirectionally and respectively fed into the next LM and GNN layer. Therefore, the GNN layer can reason on the subgraph with contextual text representation to enrich the graph node representation, and the LM layer can further encode the text with the graph node representation to improve the text representation. We adopt a multi-level connection structure to connect each layer, enabling text and graph representation with different semantic levels to interact, generating a more comprehensive feature representation. Meanwhile, we propose a QA-aware node based KG subgraph construction method. We use a question-aware node and an answer-aware node to aggregate the question entity nodes and the answer entity nodes respectively and then guide the expansion and construction process of the subgraph to enhance the connectivity of the subgraph and reduce the introduction of noise. Moreover, the QA-aware node can help the model perceive the difference between different types of nodes and help the model to learn the representation of graph nodes better.

The main contributions of this work are summarized as follows:

- We propose CORN, which adopts a bidirectional multi-level connection structure. CORN uses Co-Attention Transformer to connect each layer of the LM and GNN, which allows LM to perceive graph information and enables GNN to integrate contextual text information, so that GNN and LM can generate richer text and graph node representations.
- We propose a QA-aware node based KG subgraph construction method. Question entity nodes and answer entity nodes are aggregated through the QA-aware nodes and then the QA-aware nodes guide the subgraph expansion and construction to improve the connectivity and reduce the noise.
- We conduct extensive experiments on OpenBookQA and CommonsenseQA, and CORN achieves state-of-the-art performance compared to other KGs+PLMs models.

2 Problem Statement

In this paper, we focus on the task of multiple-choice question answering which required extra knowledge of reasoning. Formally, giving a question q, a set of answer choices $\mathcal{C}$ and external knowledge graph, our purpose is to identify the correct answer from $\mathcal{C}$.

To be specific, we calculate the probability score between q and each answer choice $a \in \mathcal{C}$ and then select the answer with the highest probability score. We construct a multi-relational subgraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ by KG (detailed in §3.1). Here $\mathcal{V}$

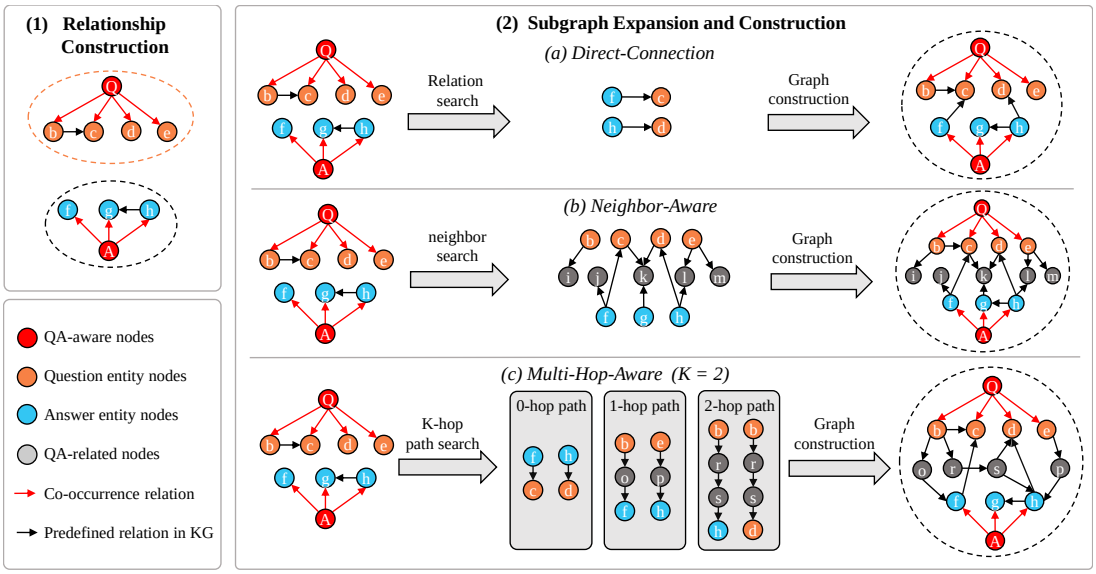

Figure 3: The QA-aware node based subgraph construction method. We first use the QA-aware nodes to aggregate QA entities and then derive different subgraph expansion strategies to construct the final subgraph.

is the subset of entity nodes from KG and is related to the mentioned entities in the QA content, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ is the set of edges that connect nodes in $\mathcal{V}$, where $\mathcal{R}$ represents a set of relation types.

3 Co-Reasoning Network

The overview of our model is shown in Figure 2. We concatenate a question q and an answer choice a to get the QA content s, where $s = [q; a]$. We apply PLMs (e.g., RoBERTa) on s to get the initial text representation $s^{'}$. For each QA content s, we use the QA-aware node based subgraph construction method (§3.1) to construct a subgraph $\mathcal{G}$ and initialize the graph node representation $\mathcal{G}^{'}$. Then, We use N-layer *Co-Reasoning Network* (CORN) to reason on the QA content $s^{'}$ and subgraph $\mathcal{G}'$. Each CORN layer consists of LM layer (§3.3), GNN layer (§3.4) and Co-Attention Transformer (§3.2), where Co-Attention Transformer bidirectionally connects LM and GNN, LM encodes QA text with graph representation, and GNN reasons on the subgraph with the contextual text representation. Finally, we use the pooled graph representation and text representation from the last CORN layer to make predictions to get the probability that the current choice a is the correct choice.

3.1 The QA-aware node based KG Subgraph Construction Method

We propose a QA-aware node based KG subgraph construction method. As shown in Figure 3, there are two stages during the process of constructing a subgraph.

Relationship Construction. We introduce a question-aware node $\mathcal{A}_q$ and an answer-aware node $\mathcal{A}_a$ which are respectively responsible for aggregating entity nodes that appear in the question context and answer context. Specifically, the question-aware node $\mathcal{A}_q$ first connects all question entity nodes $\mathcal{V}_q$ existing in the KG with the "co-occurrence" relationship, then queries the relationship in KG between each pair of the question entity nodes and connects them with the queried relationship. We construct the relationship between answer-aware node $\mathcal{A}_a$ and answer entity nodes $\mathcal{V}_a$ in the same steps.

Subgraph Expansion and Construction. After constructing relationship in QA entity nodes, we expand the subgraph to supplement additional QA-related knowledge nodes $\mathcal{V}_o$ to obtain richer graph information. To reduce the introduction of noisy nodes, we propose different subgraph expansion strategies. (a) **Direct-Connection**: This strategy does not introduce additional knowledge nodes. It only connects question-answer entity pairs that have a relationship in KG. It sacrifices some graph information in exchange for introducing the least number of noisy nodes. (b) **Neighbor-Aware**: This strategy introduces the neighbor entity nodes of question and answer entity nodes as additional knowledge nodes. It introduces rich neighbor information for each question answering entity node. (c) **Multi-Hop-Aware**: This strategy searches for

a reachable path within K-hop in KG between two nodes of question and answer entity nodes set, and introduces the nodes on the path as the additional knowledge nodes. It introduces less graph information and noisy nodes. We take different strategies to introduce additional nodes to form the subgraph $\mathcal{G}$, and evaluate each subgraph construction strategy in the experiment. We initialize the node embedding of $\mathcal{V}_q$, $\mathcal{V}_a$ and $\mathcal{V}_o$ by its entity embedding (§4.2), and simply initialize $\mathcal{A}_q$ and $\mathcal{A}_a$ with zero vectors.

3.2 Co-Attention Transformer

The most central problem of the current model is that the GNN and LM are treated as independent modules for reasoning. GNN model can not effectively use QA contextual text representation and only rely on the subgraph extracted from KGs for reasoning. Also, LM only encodes QA text and ignores the QA entity relationship. To address this problem, we bidirectionally connect each layer of GNN and LM through the Co-Attention Transformer, which allows these two modules to interact with each other's information to improve the text and graph node representation. The structure of the Co-Attention Transformer is shown in Figure 2 (b).

Specially, given the text representation $H_t^{(l)} \in \mathbb{R}^{m\times d}$ from the l-th layer of LM and node representation $H_g^{(l)} \in \mathbb{R}^{n\times d}$ from the l-th layer of GNN model, where m, n are the text length and number of nodes and d is the hidden size, we map them to query Q^t, Q^g, key K^t, K^g and value V^t, V^g matrices as in a standard transformer layer:

$$\begin{aligned} Q_i &= H^{(l)} W_i^Q \\ K_i &= H^{(l)} W_i^K \\ V_i &= H^{(l)} W_i^V \end{aligned} \tag{1}$$

where i is i-th of h matrices, $\{W_i^Q, W_i^K, W_i^V\} \in \mathbb{R}^{d\times d_k}$ are parameter matrices and $d_k \in \mathbb{R}^{d/h}$.

Then, we apply two transformer layers and exchange key-value pairs in multi-head attention to perform interactive computation. Q^t, K^g and V^g form one group and Q^g, K^t and V^t form other group. Each group performs multi-head attention computation conditioned on the other modules. The single attention head is as following:

$$\mathrm{T_i}(Q_i^t, K_i^g, V_i^g) = \mathrm{softmax}\left(\frac{Q_i^t K_i^{gT}}{\sqrt{d_k}}\right) V_i^g \tag{2}$$

$$\mathrm{G_i}(Q_i^g, K_i^t, V_i^t) = \mathrm{softmax}\left(\frac{Q_i^g K_i^{tT}}{\sqrt{d_k}}\right) V_i^t \tag{3}$$

The attention outputs of each head are then concatenated and followed by a linear transformation as following:

$$\begin{aligned} O_t^{(l)} &= \mathrm{MultiHead}(Q^t, K^g, V^g) \\ &= \mathrm{Concat}\left(\mathrm{T_1}, \cdots, \mathrm{T_h}\right) W_t^O \end{aligned} \tag{4}$$

$$\begin{aligned} O_g^{(l)} &= \mathrm{MultiHead}(Q^g, K^t, V^t) \\ &= \mathrm{Concat}\left(\mathrm{G_1}, \cdots, \mathrm{G_h}\right) W_g^O \end{aligned} \tag{5}$$

where $\{W_t^O, W_g^O\} \in \mathbb{R}^{hd_k\times d}$ are parameter matrices, $O_t^{(l)} \in \mathbb{R}^{m\times d}$ and $O_g^{(l)} \in \mathbb{R}^{n\times d}$. After that, two residual add operations are worked on the initial representation and output of multi-head attention to get a fused representation of text and graph:

$$H_t^{(l)} = \mathrm{LayerNorm}(H_t^{(l)} + O_t^{(l)}) \tag{6}$$

$$H_g^{(l)} = \mathrm{LayerNorm}(H_g^{(l)} + O_g^{(l)}) \tag{7}$$

where LayerNorm is the layer normalization operation (Ba et al., 2016). $H_t^{(l)}$ is the text representation with graph information and $H_g^{(l)}$ is the node representation with text information. Then, two feed forward networks (MLP) and two another residual add operations are applied on the above representation to get the Co-Attention Transformer output:

$$H_t^{(l+1)} = \mathrm{LayerNorm}(H_t^{(l)} + \mathrm{MLP}(H_t^{(l)})) \tag{8}$$

$$H_g^{(l+1)} = \mathrm{LayerNorm}(H_g^{(l)} + \mathrm{MLP}(H_g^{(l)})) \tag{9}$$

where $H_t^{(l+1)} \in \mathbb{R}^{m\times d}$ and $H_g^{(l+1)} \in \mathbb{R}^{n\times d}$ are the input of $(i+1)$-th layer of LM and GNN. We use Co-Attention Transformer to connect each layer of the LM and GNN layer, which can fuse semantic features of different levels to obtain a more comprehensive representation.

3.3 Language Model

To effectively utilize the capability of the PLM, we do not modify its architecture and use it to encode the text at first. Specifically, we apply PLM (e.g., RoBERTa) on the QA content s to get initial text representation $H_t^{(0)}$:

$$H_t^{(0)} = \mathrm{PLM}(s), \tag{10}$$

where $H_t^{(0)} \in \mathbb{R}^{m\times d_p}$, m is the text length and d_p is the hidden size of PLM.

Before Co-Reasoning Network, we use an MLP to unify the hidden size:

$$H_t^{(0)} = \mathrm{MLP}(H_t^{(0)}), \tag{11}$$

where the new $H_t^{(0)} \in \mathbb{R}^{m\times d}$, d is the unified hidden size.

After that, we use an N-layer Co-Reasoning Network which consists of LM, GNN, and Co-Attention Transformer for co-reasoning. For the l-th layer, the input text representation $H_t^{(l-1)}$ interacts with node representation $H_g^{(l-1)}$ in graph (detailed in §3.4) through Co-Attention Transformer and gets the text representation $H_t^{(l)}$ that is contained graph node information.

Further, we apply transformer encoder layer (Vaswani et al., 2017) as the LM in Co-Reasoning Network for reasoning:

$$H_t^{(l)} = \text{Transformer}(H_t^{(l)}) \tag{12}$$

The transformer encoder layer can encode the text representation with graph information to get the output $H_t^{(l)}$.

3.4 GNN Model

After getting knowledge concept graph $\mathcal{G}$ and initializing the entity node embedding $H_g^{(0)} \in \mathbb{R}^{n\times d_{in}}$, where n is the number of nodes and d_{in} is the initial hidden size, we also use an MLP to unify the hidden size of node:

$$H_g^{(0)} = \text{MLP}(H_g^{(0)}), \tag{13}$$

where the new $H_g^{(0)} \in \mathbb{R}^{n\times d}$, d is the unified hidden size.

We put the graph into N-layer Co-Reasoning Network. For the l-th layer, the input node representation $H_g^{(l-1)}$ interacts with text representation $H_t^{(l-1)}$ through Co-Attention Transformer and gets the node representation $H_g^{(l)}$ that is contained contextual text information.

Further, We apply RGCN (Schlichtkrull et al., 2018), a graph encoder that can encode multi-relational graphs by aggregating messages from its neighbors, as the GNN model in Co-Reasoning Network for reasoning. Specially, for each node $h_i^{(l)} \in \mathbb{R}^d$ in graph, where $[h_1^{(l)}; \cdots; h_n^{(l)}] = H_g^{(l)}$, the node representation is updated via message passing from neighbors:

$$h_i^{(l+1)} = \sigma(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}), \tag{14}$$

where $\mathcal{N}_i^r$ denotes the set of neighbor indices of node i under relation $r \in \mathcal{R}$, $c_{i,r} = |\mathcal{N}_i^r|$ is a normalization constant. $W_r^{(l)}$ is the parameter matrix related to relation r and $W_0^{(l)}$ is the parameter matrix of node i information transformation. However, the number of parameters grows rapidly with the increase in the number of relations. We apply basis decomposition to regularize the weights of R-GCN-layers:

$$W_r^{(l)} = \sum_{b=1}^{B} a_{rb}^{(l)} V_b^{(l)}, \tag{15}$$

where $V_b^{(l)} \in \mathbb{R}^{d^{(l+1)}\times d^l}$ is the parameter matrices for all relation and $a_{rb}^{(l)}$ is the coefficients depend on r. After apply RGCN-layer, the new node representations are updated and get the output $H_g^{(l)} = [h_1^{(l+1)}; \cdots; h_n^{(l+1)}]$.

3.5 Inference & Learning

The probability score for answer a as the correct answer for question q is calculated by text representation and graph node representations from the last layer:

$$p(q,a) = \text{MLP}(\text{Pool}(H_t^{(L)}) \oplus \text{Pool}(H_g^{(L)})) \tag{16}$$

where Pool is the mean pooling operation over the text representations and the node representations.

In the training process, each question provides a list of answer choices, one of which is correct. we use the cross-entropy loss function to optimize the model.

4 Experiments

4.1 Datasets

We evaluate our model on two multiple-choice question answering datasets that require external knowledge to arrive at the correct answer: OpenBookQA (Mihaylov et al., 2018) and CommonsenseQA (Talmor et al., 2019).

OpenBookQA is a multiple choice question QA task with 4 choices that require elementary science knowledge for reasoning. This dataset also provides external knowledge called *Open Books* describing scientific facts to help models answer questions. As our study focuses on reasoning by using structured knowledge, we do not utilize *Open Books* and instead utilize *ConceptNet* as the external knowledge.

CommonsenseQA is a multiple choice question QA task with 5 choices that require commonsense knowledge for reasoning. Questions and answers

are generated according to entities in *ConceptNet* and their relations. We perform experiments on the in-house (IH) data splits used in Lin et al. (2019).

4.2 Knowledge Graph

We use *ConceptNet*, a general-domain structured knowledge graph, as our external commonsense knowledge. We use the entity embeddings prepared by Feng et al. (2020), which they utilize TransE model (Bordes et al., 2013) for node embedding (100-dimensional) in *ConceptNet*. Following the work by Lin et al. (2019), we merge the original 42 relation types in *ConceptNet* into 17 relation types. The subgraph construction method for each question and answer is described in section §3.1.

4.3 Implementation & training details

We set the dimension ($D = 100$) and number of layers ($L = 3$) of our Co-Reasoning Network, with dropout rate 0.2 applied to each layer. We train the model with the RAdam optimizer using one GPU (Tesla T4). We use batch size of 64 (mini batch of 2), with 14 epochs (~4 hours) for OpenBookQA and 10 epochs (~6 hours) for CommonsenseQA.

4.4 Baseline Models

The purpose of our work is to leverage structured external knowledge for reasoning on knowledge question answering tasks. Therefore, we only compare with the models that combine PLMs and KGs, not the models using other formats of external knowledge (e.g., Wikipedia, human-annotated evidence.)

RoBERTa (Liu et al., 2019) is used as the baseline model to study the performance of PLMs without introducing extra KG information.

GconAttn (Wang et al., 2019) generalizes the Match-LSTM model in the field of text matching to knowledge concept matching.

KagNet (Lin et al., 2019) extracts the QA-related subgraph from KG, and applies GCN and LSTM to model the relational paths.

Relation Network (RN) (Santoro et al., 2017) utilizes multilayer perceptron to encode triplets on paths in KG and all the triplets representation as to the graph representation for classification.

MHGRN (Feng et al., 2020) designs a multi-hop relational reasoning module to obtain a path-level graph representation, and combines GNN and PLMs for classification.

QA-GNN (Yasunaga et al., 2021) introduces a QA content node in the subgraph for joint reasoning over the QA content and KG.

Methods	**Dev Acc.**(%)	**Test Acc.**(%)
RoBERTa-large	66.76 (±1.14)	64.80 (±2.37)
+ GconAttn	66.85 (±1.82)	64.75 (±1.48)
+ RN	67.00 (±0.71)	65.20 (±1.18)
+ MHGRN	68.10 (±1.12)	66.85 (±1.19)
+ QA-GNN	68.27 (±1.09)	67.80 (±2.75)
+ CORN (**Ours**)	**72.35** (±0.86)	**71.30** (±0.64)

Table 1: **Dev and Test accuracy on OpenBookQA.**

Methods	**IHdev-Acc.**(%)	**IHtest-Acc.**(%)
RoBERTa-large	73.07 (±0.45)	68.69 (±0.56)
+ GconAttn	72.61 (±0.39)	68.59 (±0.96)
+ KagNet	73.47 (±0.22)	69.01 (±0.76)
+ RN	74.57 (±0.91)	69.08 (±0.21)
+ MHGRN	74.45 (±0.10)	71.11 (±0.81)
+ QA-GNN	76.54 (±0.21)	73.41 (±0.92)
+ CORN (**Ours**)	**79.58** (±0.38)	**74.43** (±0.59)

Table 2: **Dev and Test accuracy on CommonsenseQA in-house split.**

4.5 Main Results

Table 1 shows the result on OpenBookQA. For fair comparison, we use the RoBERTa-large as the text encoder for all models. Our model achieves the best performance across all baseline models by greatly improving the dev accuracy by ~4.08% and test accuracy by ~3.5%. The improvement over QA-GNN suggests that CORN is a better method to combine the QA content and KG for co-reasoning. Notably, QA-GNN (1.1B total parameters) uses 2.5x more total parameters than our model (our has 440M total parameters). This benefits from CORN's use of fewer layers, smaller hidden dimension, and simpler designed GNN but more efficient connection structure. In addition, we did not compare with other models that use the extra corpus of scientific facts provided by official, because our purpose is to reason from a structured knowledge graph.

Table 2 shows the result on CommonsenseQA. All models also use Roberta-large as the text encoder. CORN achieves state-of-art performance across all existing models with improving the dev accuracy by ~3.04% and test accuracy by ~1.02%. The result suggests that CORN improves the performance of through the bidirectional interaction

Methods	OpenBookQA	CommonseQA
Direct-Connection	69.27 (±0.41)	**74.43** (±0.59)
Neighbor-Aware	**71.30** (±0.64)	72.66 (±0.40)
Multi-Hop-Aware (K=1)	69.60 (±1.28)	71.64 (±0.16)
Multi-Hop-Aware (K=2)	68.30 (±1.30)	71.77 (±0.56)

Table 3: **Results of different subgraph construction methods.** We report the test accuracy on OpenBookQA and CommonsenseQA.

Method	Test Acc.
CORN	**71.30**
w/o CAT (Text)	70.17
w/o CAT (Graph)	70.20
w/o CAT (Multi-level)	70.26
w/o CAT	69.67
w/o QA-aware nodes	70.20

CORN Layers	Test Acc.
$L = 2$	70.90
$L = 3$	**71.30**
$L = 4$	70.73
$L = 5$	70.40

Table 4: **Ablation study** of our model components, using the OpenBookQA test set. CAT is the abbreviation of Co-Attention Transformer.

of GNN and LM without designing complex graph inference networks.

4.6 Subgraph Construction Result

Table 3 shows the results of constructing subgraphs by different subgraph expansion methods. We evaluate our proposed strategies for introducing additional QA-related knowledge nodes in KG.

For OpenBookQA, we find that the *Neighbor-Aware* performs best. The OpenBookQA emphasizes reasoning using multiple scientific knowledge. The question entity nodes require multiple scientific knowledge to connect with the answer entity nodes. Therefore, the model cannot perform reasoning efficiently without introducing extra nodes, which is the reason that *Direct-Connect* performs worst. Compared to *Multi-Hop-Aware (K=1)* and *Multi-Hop-Aware (K=2)*, *Neighbor-Aware* can provide extra influential related entity nodes to cover possible scientific knowledge, further helping the model for reasoning.

For CommonsenseQA, the best strategy is to not introduce additional QA-related knowledge nodes (*Direct-Connection*). We analyze that this is caused by the construction method of CommonsenseQA. The four answer entities and one question entity in CommonsenseQA are directly connected in ConceptNet. Therefore, the introduction of additional QA-related knowledge nodes will lead to noise and redundancy, which will reduce the performance of the model.

4.7 Ablation Study

The ablation study on each of our model components is shown in Table 4, using the OpenBookQA test set. We found that removing the attention computation of any module (text or graph) in the Co-Attention Transformer would result in a performance drop of ~1.1%. We also test removing the multi-level connection structure and only keeping one layer of the Co-Attention Transformer, the degraded performance result shows that multi-level connections can indeed interact with different levels of semantic features to get richer representation. Removing Co-Attention Transformer will significantly degrade performance by 1.63%, which proves the importance of connecting and co-reasoning between LM and GNN. We also analyze the impact of QA-aware nodes. When we remove QA-aware nodes, the performance of the model drops by 1.1%, which proves that QA-aware nodes can help the model to perform better reasoning. For the number of CORN layers, we find $L = 3$ works best on the dev set, which is also similar to the number of layers generally used in GNN.

4.8 Model Visualization

The purpose of our model is to make GNN and LM mutually aware of the information of each other's modules for reasoning. Therefore, we analyze the attention weights of text module and graph module in Co-Attention Transformer. Figure 4 gives a visualization of an example. Given question *"Where would you find magazines along side many other printed works?"* and choices *"A. doctor B. bookstore"*, we show the attention weights of the last Co-Attention Transformer layer under these two choices separately. The key entity in this question is *"magazines"*. For the wrong choice *"doctor"*, though the attention of text module can give a higher weight of *"magazines"* entity in the graph, the attention weight distribution of graph module is rather average, and cannot provide meaningful information. For the correct choice *"bookstore"*, not only the attention of text module can capture the importance of the *"magazine"* entity in the graph, but also the attention in graph module gives a higher weight to the *"bookstore"* in the text, which is also the correct answer. Therefore, the Co-Attention Transformer in CORN can effectively capture the relationship between the QA text and knowledge graph formed by the correct choice.

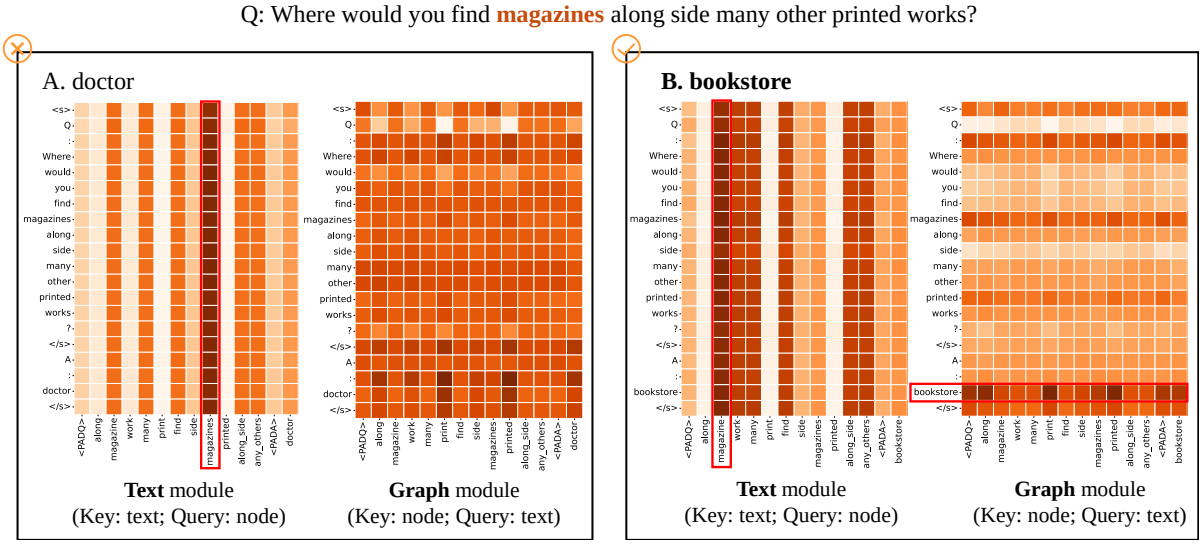

Figure 4: **Visualization of the attention weights of the last CORN layer.** Given a question and corresponding choices. We show the attention weights between text and graph formed by each choice. The row indices of the heatmap are the words in the text, and the column indices are the entity nodes in the graph. The red box represents the part with the highest attention weight.

5 Related Work

Question answering with PLMs. The recent success of PLMs in various NLP tasks has prompted much work to try directly utilize PLMs to encode external knowledge. These work can be divided into two paradigms: 1) Format external knowledge (eg. Wikipedia, knowledge graph) into text or triples as corpus for PLM pre-training task (Ye et al., 2019; Li et al., 2019; Sun et al., 2019; Gururangan et al., 2020). 2) Fine-tuning PLMs with evidence for external knowledge (Pan et al., 2019; Lv et al., 2020). However, such models can not provide interpretable reasoning process, which is the key to commonsense reasoning.

Question answering with KG+LM. Many works attempt to additionally perform reasoning with GNN on knowledge graphs to address the problem that PLMs unable to reason on structured knowledge. GCN (Kipf and Welling, 2017) aggregates the neighborhood information of each node for message passing. RGCN (Schlichtkrull et al., 2018) can encode multi-relational graphs by aggregating messages from its neighbors of different relations. GAT (Velickovic et al., 2018) assigns different attention weights to aggregate each node feature, and is used (Chen et al., 2019) to distinguish the importance of different concept entity nodes in KG. The related work of question answering (Lin et al., 2019; Feng et al., 2020; Lv et al., 2020) try to design complex graph neural networks for single-hop or multi-hop reasoning in KG. However, these works treat the QA content and KG as separate modules. Though Yasunaga et al. (2021) add the QA content to graph for joint reasoning, it still cannot solve the problem that information unable exchange between GNN and LM. CORN addresses the above problem by connecting each layer of these two models through Co-Attention Transformer for co-reasoning.

6 Conclusion

We propose a novel commonsense QA model, CORN, which adopts a bidirectional multi-level connection structure. It bidirectionally connects each layer of the LM and GNN through the Co-Attention Transformer, which enables the LM to perceive the relationship of QA entity nodes to improve the text representation, and allows GNN to utilize contextual text information to enhance the graph node representation. Meanwhile, we propose a QA-aware node based KG subgraph construction method. The QA-aware nodes aggregate question entity nodes and answer entity nodes and then guide the subgraph expansion and construction to increase the connectivity of the subgraph, and reduce the introduction of noise. Through extensive experiments and visual analysis, CORN can perform multi-level bidirectional interaction to improve the LM+KG models, and achieves state-of-the-art performance among them.

Acknowledgements

This work is supported by National Key R&D Project of China under Grants No.2021QY2102. National Natural Science Foundation of China under Grants No.62172089, No.61972087, No.62172090, No.62106045. Natural Science Foundation of Jiangsu Province under Grants No.BK20191258. Jiangsu Provincial Key Laboratory of Computer Networking Technology. Jiangsu Provincial Key Laboratory of Network and Information Security under Grants No.BM2003201, and Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grants No.93K-9, Nanjing Purple Mountain Laboratory.

References

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250. ACM.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2787–2795.

Daoyuan Chen, Yaliang Li, Min Yang, Hai-Tao Zheng, and Ying Shen. 2019. Knowledge-aware textual entailment with graph attention network. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 2145–2148. ACM.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186. Association for Computational Linguistics.

Yanlin Feng, Xinyue Chen, Bill Yuchen Lin, Peifeng Wang, Jun Yan, and Xiang Ren. 2020. Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1295–1309. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8342–8360. Association for Computational Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Shiyang Li, Jianshu Chen, and Dian Yu. 2019. Teaching pretrained models with commonsense reasoning: A preliminary kb-based approach. *CoRR*, abs/1909.09743.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. Kagnet: Knowledge-aware graph networks for commonsense reasoning. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 2829–2839. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8449–8456. AAAI Press.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2381–2391. Association for Computational Linguistics.

Xiaoman Pan, Kai Sun, Dian Yu, Jianshu Chen, Heng Ji, Claire Cardie, and Dong Yu. 2019. Improving question answering with external knowledge. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering, MRQA@EMNLP 2019, Hong Kong, China, November 4, 2019*, pages 27–37. Association for Computational Linguistics.

Adam Santoro, David Raposo, David G. T. Barrett, Mateusz Malinowski, Razvan Pascanu, Peter W. Battaglia, and Tim Lillicrap. 2017. A simple neural network module for relational reasoning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4967–4976.

Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings*, volume 10843 of *Lecture Notes in Computer Science*, pages 593–607. Springer.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 4444–4451. AAAI Press.

Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. ERNIE: enhanced representation through knowledge integration. *CoRR*, abs/1904.09223.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4149–4158. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Peifeng Wang, Nanyun Peng, Filip Ilievski, Pedro A. Szekely, and Xiang Ren. 2020. Connecting the dots: A knowledgeable path generator for commonsense question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4129–4140. Association for Computational Linguistics.

Xiaoyan Wang, Pavan Kapanipathi, Ryan Musa, Mo Yu, Kartik Talamadupula, Ibrahim Abdelaziz, Maria Chang, Achille Fokoue, Bassem Makni, Nicholas Mattei, and Michael Witbrock. 2019. Improving natural language inference using external knowledge in the science questions domain. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7208–7215. AAAI Press.

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, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 535–546. Association for Computational Linguistics.

Zhi-Xiu Ye, Qian Chen, Wen Wang, and Zhen-Hua Ling. 2019. Align, mask and select: A simple method for incorporating commonsense knowledge into language representation models. *CoRR*, abs/1908.06725.