GREASELM: GRAPH REASONING ENHANCED LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Answering complex questions about textual narratives requires reasoning over both stated context and the world knowledge that underlies it. However, pretrained language models (LM), the foundation of most modern QA systems, do not robustly represent latent relationships between concepts, which is necessary for reasoning. While knowledge graphs (KG) are often used to augment LMs with structured representations of world knowledge, it remains an open question how to effectively fuse and reason over the KG representations and the language context, which provides situational constraints and nuances. In this work, we propose GREASELM, a new model that fuses encoded representations from pretrained LMs and graph neural networks over multiple layers of modality interaction operations. Information from both modalities propagates to the other, allowing language context representations to be grounded by structured world knowledge, and allowing linguistic nuances (*e.g.*, negation, hedging) in the context to inform the graph representations of knowledge. Our results on three benchmarks in the commonsense reasoning (*i.e.*, CommonsenseQA, OpenbookQA) and medical question answering (*i.e.*, MedQA-USMLE) domains demonstrate that GREASELM can more reliably answer questions that require reasoning over both situational constraints and structured knowledge, even outperforming models 8× larger.

1 INTRODUCTION

Question answering is a challenging task that requires complex reasoning over both explicit constraints described in the textual context of the question, as well as unstated, relevant knowledge about the world (*i.e.*, knowledge about the domain of interest). Recently, large pretrained language models fine-tuned on QA datasets have become the dominant paradigm in NLP for question answering tasks (Khashabi et al., 2020). After pretraining on an extreme-scale collection of general text corpora, these language models learn to implicitly encode broad knowledge about the world, which they are able to leverage when fine-tuned on a domain-specific downstream QA task. However, despite the strong performance of this two-stage learning procedure on common benchmarks, these models struggle when given examples that are distributionally different from examples seen during fine-tuning (McCoy et al., 2019). Their learned behavior often relies on simple (at times spurious) patterns to offer shortcuts to an answer, rather than robust, structured reasoning that effectively fuses the explicit information provided by the context and implicit external knowledge (Marcus, 2018).

On the other hand, massive knowledge graphs (KG), such as Freebase (Bollacker et al., 2008), Wikidata (Vrandečić & Krötzsch, 2014), ConceptNet (Speer et al., 2017), and Yago (Suchanek et al., 2007) capture such external knowledge explicitly using triplets that capture relationships between entities. Previous research has demonstrated the significant role KGs can play in structured reasoning and query answering (Ren et al., 2020; Ren & Leskovec, 2020). However, extending these reasoning advantages to general QA (where questions and answers are expressed in natural language and not easily mapped to strict logical queries) requires finding the right integration of knowledge from the KG with the information and constraints provided by the QA example. Prior methods propose various ways to leverage both modalities (*i.e.*, expressive large language models and structured KGs) for improved reasoning (Mihaylov & Frank, 2018; Lin et al., 2019; Feng et al., 2020). However, these methods typically fuse the two modalities in a shallow and non-interactive manner, encoding both separately and fusing them at the output for a prediction, or using one to augment the input of the other. Consequently, previous methods demonstrate restricted capacity to exchange

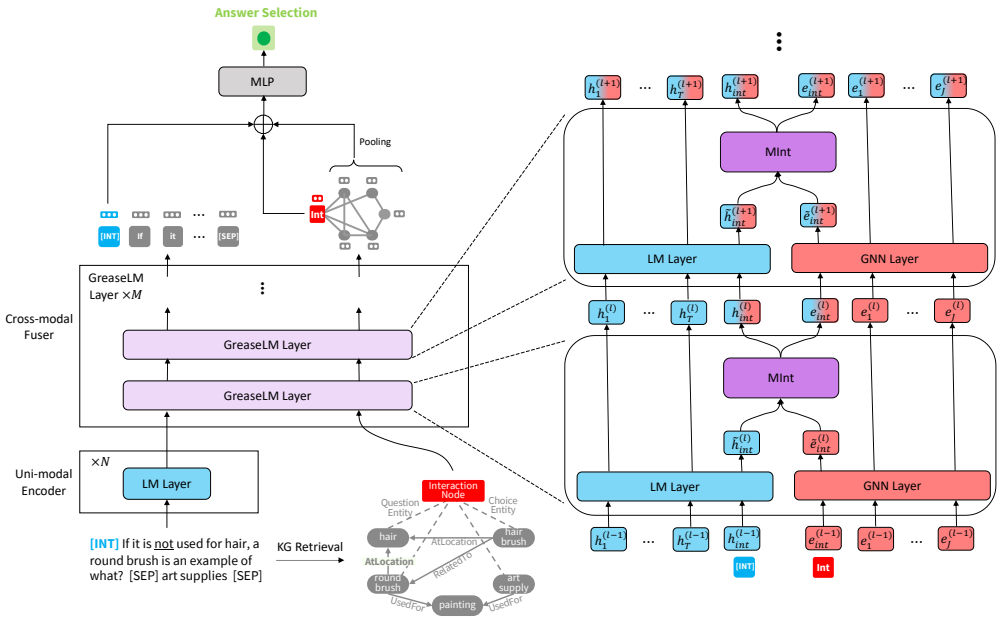

Figure 1: **GREASELM Architecture**. The textual context is appended with a special *interaction token* and passed through N LM-based unimodal encoding layers. Simultaneously, a local KG of relevant knowledge is extracted and connected to an *interaction node*. In the later GREASELM layers, the language representation continues to be updated through LM layers and the KG is processed using a GNN, simulating reasoning over its knowledge. In each layer, after each modality's representation is updated, the representations of the *interaction token* and *node* are pulled, concatenated, and passed through a modality interaction (MInt) unit to mix their representations. In subsequent layers, the mixed information from the *interaction* elements mixes with their respective modalities, allowing knowledge from the KG to affect the representations of individual tokens, and context from language to affect fine-grained entity knowledge representations in the GNN.

useful information between the two modalities. It remains an open question how to effectively fuse the KG and LM representations in a truly unified manner, where the two representations can interact in a non-shallow way to simulate structured, situational reasoning.

In this work, we present GREASELM, a new model that enables fusion and exchange of information from both the LM and KG in multiple layers of its architecture (see Figure 1). Our proposed GREASELM consists of an LM that takes as input the natural language context, as well as a graph neural network (GNN) that reasons over the KG. After each layer of the LM and GNN, we design an interactive scheme to bidirectionally transfer the information from each modality to the other through specially initialized interaction representations (*i.e.*, *interaction* token for the LM; *interaction* node for the GNN). In such a way, all the tokens in the language context receive information from the KG entities through the interaction token and the KG entities indirectly interact with the tokens through the interaction node. By such a deep integration across all layers, GREASELM enables joint reasoning over both the language context and the KG entities under a unified framework agnostic to the specific language model or graph neural network, so that both modalities can be contextualized by the other.

GREASELM demonstrates significant performance gains across different LM architectures. We perform experiments on several standard QA benchmarks: CommonsenseQA, OpenbookQA and MedQA-USMLE, which require external knowledge across different domains (commonsense reasoning and medical reasoning) and use different KGs (ConceptNet and Disease Database). Across both domains, GREASELM outperforms comparably-sized prior QA models, including strong fine-tuned LM baselines (by 5.5%, 6.6%, and 1.3%, respectively) and state-of-the-art KG+LM models (by 0.9%, 1.8%, and 0.5%, respectively) on the three competitive benchmarks. Furthermore, with the deep fusion of both modalities, GREASELM exhibits strong performance over baselines on questions that exhibit textual nuance, such as resolving multiple constraints, negation, and hedges, and which require effective reasoning over both language context and KG.

2 RELATED WORK

Integrating KG information has become a popular research area for improving neural QA systems. Some works explore using two-tower models to answer questions, where a graph representation of knowledge and language representation are fused with no interaction between them (Wang et al., 2019). Other works seek to use one modality to ground the other, such as using an encoded representation of a linked KG to augment the textual representation of a QA example (*e.g.*, Knowledgeable Reader, Mihaylov & Frank, 2018; KagNet, Lin et al., 2019; KT-NET Yang et al., 2019). Others reverse the flow of information and use a representation of the text (*e.g.*, final layer of LM) to provide an augmentation to a graph reasoning model over an extracted KG for the example (*e.g.*, MHGRN, Feng et al., 2020; Lv et al., 2020). In all of these settings, however, the interaction between both modalities is limited as information between them only flows one way.

More recent approaches explore deeper integrations of both modalities. Certain approaches learn to access implicit knowledge encoded in LMs (Bosselut et al., 2019; Petroni et al., 2019; Hwang et al., 2021) by training on structured KG data, and then use the LM to generate local KGs that can be used for QA (Wang et al., 2020; Bosselut et al., 2021). However, these approaches discard the static KG once they train the LM on its facts, losing important structure that can guide reasoning. More recently, QA-GNN (Yasunaga et al., 2021) proposed to jointly update the LM and GNN representations via message passing. However, they use a single pooled representation of the LM to seed the textual component of this joint structure, limiting the updates that can be made to the textual representation. In contrast to prior works, we propose to make individual token representations in the LM and node representations in the GNN mix for multiple layers, enabling representations of both modalities to reflect particularities of the other (e.g., knowledge grounds language; language nuances specifies which knowledge is important). Simultaneously, we retain the individual structure of both modalities, which we demonstrate improves QA performance substantially (§5).

3 PROPOSED APPROACH: GREASELM

In this work, we augment large-scale language models (Devlin et al., 2019; Liu et al., 2019; Lan et al., 2020; Liu et al., 2020) with graph reasoning modules over KGs. Our method, GREASELM (depicted in Figure 1), consists of two stacked components: (1) a set of unimodal LM layers which learn an initial representation of the input tokens, and (2) a set of upper cross-modal GREASELM layers which learn to jointly represent the language sequence and linked knowledge graph, allowing textual representations formed from the underlying LM layers and a graph representation of the KG to mix with one another. We denote the number of LM layers as N, and the number of GREASELM layers as M. The total number of layers in our model is $N + M$.

Notation In the task of *multiple choice question answering (MCQA)*, a generic MCQA-type dataset consists of examples with a context paragraph c, a question q and a candidate answer set $\mathcal{A}$, all expressed in text. In this work, we also assume access to an external knowledge graph (KG) $\mathcal{G}$ that provides background knowledge that is relevant to the content of the multiple choice questions.

Given a QA example $(c, q, \mathcal{A})$, and the KG $\mathcal{G}$ as input, our goal is to identify which answer $a \in \mathcal{A}$ is correct. Without loss of generality, when an operation is applied to an arbitrary answer, we refer to that answer as a. We denote a sequence of tokens in natural language as $\{w_1, \ldots, w_T\}$, where T is the total number of tokens, and the representation of a token w_t from the ℓ-th layer of the model as $\boldsymbol{h}_t^{(\ell)}$. We denote a set of nodes as $\{e_1, \ldots, e_J\}$, where J is the total number of nodes, and the representation of a node e_j in the ℓ-th layer of the model as $\boldsymbol{e}_j^{(\ell)}$.

3.1 INPUT REPRESENTATION

We concatenate our context paragraph c, question q, and candidate answer a with separator tokens to get our model input $[c; q; a]$ and tokenize the combined sequence into $\{w_1, \ldots, w_T\}$. Second, we use the input sequence to retrieve a subgraph of the KG $\mathcal{G}$ (denoted $\mathcal{G}_{\text{sub}}$), which provides knowledge from the KG that is relevant to this QA example. We denote the set of nodes in $\mathcal{G}_{\text{sub}}$ as $\{e_1, \ldots, e_J\}$.

KG retrieval. Given each QA context, we follow the procedure from Yasunaga et al. (2021) to retrieve the subgraph $\mathcal{G}_{\text{sub}}$ from $\mathcal{G}$. First, we perform entity linking to $\mathcal{G}$ to retrieve an initial set of

nodes $\mathcal{V}_{\text{linked}}$. Second, we add any bridge entities that are in a 2-hop path between any pair of linked entities in $\mathcal{V}_{\text{linked}}$ to get the final set of retrieved entities $\mathcal{V}_{\text{sub}}$. We then retrieved all the edges that connect any two nodes in $\mathcal{V}_{\text{sub}}$, forming the retrieved subgraph $\mathcal{G}_{\text{sub}}$. Each node is assigned a type according to whether its corresponding entity was linked from the context c, question q, answer a, or from a bridge path. In the remaining parts of the paper, we use the term "KG" to refer to $\mathcal{G}_{\text{sub}}$.

Interaction Bottlenecks. In the cross-modal GREASELM layers, information is fused between both modalities, for which we define a special *interaction token* w_{int} and a special *interaction node* e_{int} whose representations serve as the bottlenecks through which the two modalities interact (§3.3). We prepend w_{int} to the token sequence and connect e_{int} to all the *linked* nodes $\mathcal{V}_{\text{linked}}$ in $\mathcal{G}_{\text{sub}}$.

3.2 LANGUAGE PRE-ENCODING

In the unimodal encoding component, given the sequence of tokens $\{w_{int}, w_1, \ldots, w_T\}$, we first sum the token, segment, and positional embeddings for each token to compute its ℓ=0 input representation $\{\boldsymbol{h}_{int}^{(0)}, \boldsymbol{h}_1^{(0)}, \ldots, \boldsymbol{h}_T^{(0)}\}$, and then compute an output representation for each layer ℓ:

$$\{\boldsymbol{h}_{int}^{(\ell)}, \boldsymbol{h}_1^{(\ell)}, \ldots, \boldsymbol{h}_T^{(\ell)}\} = \text{LM-Layer}(\{\boldsymbol{h}_{int}^{(\ell-1)}, \boldsymbol{h}_1^{(\ell-1)}, \ldots, \boldsymbol{h}_T^{(\ell-1)}\}) \quad \text{for } \ell = 1, \ldots, N \tag{1}$$

where LM-Layer($\cdot$) is a single LM encoder layer, whose parameters are initialized using a pretrained model (§4.1). We refer readers to Vaswani et al. (2017) for technical details of these layers.

3.3 GREASELM

GREASELM uses a cross-modal fusion component to inject information from the KG into language representations and information from language into KG representations. The GREASELM layer is designed to separately encode information from both modalities, and fuse their representations using the bottleneck of the special interaction token and node. It is comprised of three components: (1) a transformer LM encoder block which continues to encode the language context, (2) a GNN layer that reasons over KG entities and relations, and (3) a modality interaction layer that takes the unimodal representations of the interaction token and interaction node and exchanges information through them. We discuss these three components below.

Language Representation. In the ℓ-th GREASELM layer, the input token embeddings $\{\boldsymbol{h}_{int}^{(N+\ell-1)}, \boldsymbol{h}_1^{(N+\ell-1)}, \ldots, \boldsymbol{h}_T^{(N+\ell-1)}\}$ are fed into additional transformer LM encoder blocks that continue to encode the textual context based on the LM's pretrained representations:

$$\{\tilde{\boldsymbol{h}}_{int}^{(N+\ell)}, \tilde{\boldsymbol{h}}_1^{(N+\ell)}, \ldots, \tilde{\boldsymbol{h}}_T^{(N+\ell)}\} = \text{LM-Layer}(\{\boldsymbol{h}_{int}^{(N+\ell-1)}, \boldsymbol{h}_1^{(N+\ell-1)}, \ldots, \boldsymbol{h}_T^{(N+\ell-1)}\}) \quad \text{for } \ell = 1, \ldots, M \tag{2}$$

where $\tilde{\boldsymbol{h}}$ corresponds to pre-fused embeddings of the language modality. As we will discuss below, because $\boldsymbol{h}_{int}^{N+\ell-1}$ will encode information received from the knowledge graph representation, these late language encoding layers will also allow the token representations to mix with KG knowledge.

Graph Representation. The GREASELM layers also encode a representation of the local KG $\mathcal{G}_{\text{sub}}$ linked from the QA example. To represent the graph, we first compute initial node embeddings $\{\boldsymbol{e}_1^{(0)}, \ldots, \boldsymbol{e}_J^{(0)}\}$ for the retrieved entities using pretrained KG embeddings for these nodes (§4.1). The initial embedding of the interaction node $\boldsymbol{e}_{int}^{0}$ is initialized randomly.

Then, in each layer of the GNN, the current representation of the node embeddings $\{\boldsymbol{e}_{int}^{(\ell-1)}, \boldsymbol{e}_1^{(\ell-1)}, \ldots, \boldsymbol{e}_J^{(\ell-1)}\}$ is fed into the layer to perform a round of information propagation between nodes in the graph and yield pre-fused node embeddings for each entity:

$$\{\tilde{\boldsymbol{e}}_{int}^{(\ell)}, \tilde{\boldsymbol{e}}_1^{(\ell)}, \ldots, \tilde{\boldsymbol{e}}_J^{(\ell)}\} = \text{GNN}(\{\boldsymbol{e}_{int}^{(\ell-1)}, \boldsymbol{e}_1^{(\ell-1)}, \ldots, \boldsymbol{e}_J^{(\ell-1)}\}) \quad \text{for } \ell = 1, \ldots, M \tag{3}$$

where GNN corresponds to a variant of graph attention networks (Veličković et al., 2018) that is a simplification of the method of Yasunaga et al. (2021). The GNN computes node representations

$\tilde{\boldsymbol{e}}_j^{(\ell-1)}$ for each node $e_j \in \{e_1, \ldots, e_J\}$ via message passing between neighbors on the graph.

$$\tilde{\boldsymbol{e}}_j^{(\ell)} = f_n\left(\sum_{e_s \in \mathcal{N}_{e_j} \cup \{e_j\}} \alpha_{sj} \boldsymbol{m}_{sj}\right) + \boldsymbol{e}_j^{(\ell-1)} \tag{4}$$

where $\mathcal{N}_{e_j}$ represents the neighborhood of an arbitrary node e_j, $\boldsymbol{m}_{sj}$ denotes the message one of its neighbors e_s passes to e_j, α_{sj} is an attention weight that scales the message $\boldsymbol{m}_{sj}$, and f_n is a 2-layer MLP. The messages $\boldsymbol{m}_{sj}$ between nodes allow entity information from a node to affect the model's representation of its neighbors, and are computed in the following manner:

$$\boldsymbol{r}_{sj} = f_r(\tilde{\boldsymbol{r}}_{sj},\ \boldsymbol{u}_s,\ \boldsymbol{u}_j) \tag{5}$$

$$\boldsymbol{m}_{sj} = f_m(\boldsymbol{e}_s^{(\ell)},\ \boldsymbol{u}_s,\ \boldsymbol{r}_{sj}) \tag{6}$$

where $\boldsymbol{u}_s, \boldsymbol{u}_j$ are node type embeddings, $\tilde{\boldsymbol{r}}_{sj}$ is a relation embedding for the relation connecting e_s and e_j, f_r is a 2-layer MLP, and f_m is a linear transformation. The attention weights α_{sj} scale the contribution of each neighbor's message by its importance, and are computed as follows:

$$\boldsymbol{q}_s = f_q(\boldsymbol{e}_s^{(\ell)},\ \boldsymbol{u}_s) \tag{7}$$

$$\boldsymbol{k}_j = f_k(\boldsymbol{e}_j^{(\ell)},\ \boldsymbol{u}_j,\ \boldsymbol{r}_{sj}) \tag{8}$$

$$\gamma_{sj} = \frac{\boldsymbol{q}_s^\top \boldsymbol{k}_j}{\sqrt{D}} \tag{9}$$

$$\alpha_{sj} = \frac{\exp(\gamma_{sj})}{\sum_{e_s \in \mathcal{N}_{e_j} \cup \{e_j\}} \exp(\gamma_{sj})} \tag{10}$$

where f_q and f_k are linear transformations and $\boldsymbol{u}_s, \boldsymbol{u}_j, \boldsymbol{r}_{sj}$ are defined the same as above.

As discussed in the following paragraph, message passing between the interaction node e_{int} and the nodes from the retrieved subgraph will allow information from text that e_{int} receives from w_{int} to propagate to the other nodes in the graph.

Modality Interaction. Finally, after using a transformer LM layer and a GNN layer to update token embeddings and node embeddings respectively, we use a *modality interaction layer* (MInt) to let the two modalities *fuse* information through the bottleneck of the interaction token w_{int} and the interaction node e_{int}. We concatenate the pre-fused embeddings of the interaction token $\tilde{\boldsymbol{h}}_{int}^{(i)}$ and interaction node $\tilde{\boldsymbol{e}}_{int}^{(i)}$, pass the joint representation through a mixing operation (MInt), and then split the output post-fused embeddings into $\boldsymbol{h}_{int}^{(i)}$ and $\boldsymbol{e}_{int}^{(i)}$:

$$[\boldsymbol{h}_{int}^{(\ell)}; \boldsymbol{e}_{int}^{(\ell)}] = \text{ MInt}([\tilde{\boldsymbol{h}}_{int}^{(\ell)}; \tilde{\boldsymbol{e}}_{int}^{(\ell)}]), \tag{11}$$

We use a two-layer MLP as our MInt operation, though other fusion operators could be used to mix the representation. All the tokens other than the interaction token w_{int} and all the nodes other than the interaction node e_{int} are not involved in the modality interaction process: $\boldsymbol{w}^{(\ell)} = \tilde{\boldsymbol{w}}^{(\ell)}$ for $w \in \{w_1, \ldots, w_T\}$ and $\boldsymbol{e}^{(\ell)} = \tilde{\boldsymbol{e}}^{(\ell)}$ for $e \in \{e_1, \ldots, e_J\}$. However, they receive information from the interaction representations $\boldsymbol{h}_{int}^{(\ell)}$ and $\boldsymbol{e}_{int}^{(\ell)}$ in the next layers of their respective modal propagation (*i.e.*, Eqs. 2, 3). Consequently, across multiple GREASELM layers, information propagates between both modalities (see Fig. 1 for visual depiction), grounding language representations to KG knowledge, and knowledge representations to contextual constraints.

Learning & Inference For the MCQA task, given a question q and an answer a from all the candidates $\mathcal{A}$, we compute the probability of a being the correct answer as $p(a \mid q, c) \propto \exp(\text{MLP}(\boldsymbol{h}_{int}^{(N+M)},\ \boldsymbol{e}_{int}^{(M)},\ \boldsymbol{g}))$, where $\boldsymbol{g}$ denotes attention-based pooling of $\{\boldsymbol{e}_j^{(M)} \mid e_j \in \{e_1, \ldots, e_J\}\}$ using $\boldsymbol{h}_{int}^{(N+M)}$ as a query. We optimize the whole model end-to-end using the cross entropy loss. At inference time, we predict the most plausible answer as $\arg\max_{a \in \mathcal{A}}\ p(a \mid q, c)$.

4 EXPERIMENTAL SETUP

We evaluate GREASELM on three diverse multiple-choice question answering datasets across two domains: *CommonsenseQA* (Talmor et al., 2019) and *OpenBookQA* (Mihaylov et al., 2018) as commonsense reasoning benchmarks, and *MedQA-USMLE* (Jin et al., 2021) as a clinical QA task.

CommonsenseQA is a 5-way multiple-choice question answering dataset of 12,102 questions that require background commonsense knowledge beyond surface language understanding. We perform our experiments using the in-house data split of Lin et al. (2019) to compare to baseline methods.

Dataset	Example
CommonsenseQA	A weasel has a thin body and short legs to easier burrow after prey in a what? (A) tree (B) mulberry bush (C) chicken coop (D) viking ship **(E) rabbit warren**
OpenbookQA	Which of these would let the most heat travel through? (A) a new pair of jeans **(B) a steel spoon in a cafeteria** (C) a cotton candy at a store (D) a calvin klein cotton hat
MedQA-USMLE	A 57-year-old man presents to his primary care physician with a 2-month history of right upper and lower extremity weakness. He noticed the weakness when he started falling far more frequently while running errands. Since then, he has had increasing difficulty with walking and lifting objects. His past medical history is significant only for well-controlled hypertension, but he says that some members of his family have had musculoskeletal problems. His right upper extremity shows forearm atrophy and depressed reflexes while his right lower extremity is hypertonic with a positive Babinski sign. Which of the following is most likely associated with the cause of this patients symptoms? (A) HLA-B8 haplotype (B) HLA-DR2 haplotype **(C) Mutation in SOD1** (D) Mutation in SMN1

Table 1: Examples of the MCQA task for each of the datasets evaluated in this work.

OpenbookQA is a 4-way multiple-choice question answering dataset that tests elementary scientific knowledge. It contains 5,957 questions along with an open book of scientific facts. We use the official data splits from Mihaylov & Frank (2018).

MedQA-USMLE is a 4-way multiple-choice question answering dataset, which requires biomedical and clinical knowledge. The questions are originally from practice tests for the United States Medical License Exams (USMLE). The dataset contains 12,723 questions. We use the original data splits from Jin et al. (2021).

4.1 Implementation & training details

Language Models. We seed GREASELM with RoBERTa-Large (Liu et al., 2019) for our experiments on CommonsenseQA, AristoRoBERTa (Clark et al., 2019) for our experiments on OpenbookQA, and SapBERT (Liu et al., 2020) for our experiments on MedQA-USMLE, demonstrating GREASELM's generality with respect to language model initializations. Hyperparameters for training these models can be found in Appendix Table 8.

Knowledge Graphs. We use *ConceptNet* (Speer et al., 2017), a general-domain knowledge graph, as our external knowledge source $\mathcal{G}$ for both CommonsenseQA and OpenbookQA. It has 799,273 nodes and 2,487,810 edges in total. To compute initial node embeddings (§3.3) for entities retrieved from ConceptNet, we follow the method of MHGRN (Feng et al., 2020). We convert knowledge triples in the KG into sentences using pre-defined templates for each relation. Then, these sentences are fed into a BERT-large LM to compute embeddings for each sentence. Finally, for all sentences containing an entity, we extract all token representations of the entity's mention spans in these sentences, mean pool over these representations and project this mean-pooled representation.

For MedQA-USMLE, we use a self-constructed knowledge graph that integrates the Disease Database portion of the Unified Medical Language System (UMLS; Bodenreider, 2004) and DrugBank (Wishart et al., 2018). The knowledge graph contains 9,958 nodes and 44,561 edges. Node embeddings are initialized using the pooled token output embeddings of the entity name from the SapBERT model (described in §4.2; Liu et al., 2020). Additional hyperparameters for preprocessing these KGs can be found in Appendix Table 8.

4.2 Baseline methods

Fine-tuned LMs. To study the effect of using KGs as external knowledge sources, we compare our method with vanilla fine-tuned LMs, which are knowledge-agnostic. We fine-tune RoBERTa-

Table 2: **Performance comparison on *CommonsenseQA* in-house split** (controlled experiments). As the official test is hidden, here we report the in-house Dev (IHdev) and Test (IHtest) accuracy, following the data split of Lin et al. (2019). Experiments are controlled using same seed LM.

Methods	**IHdev-Acc.** (%)	**IHtest-Acc.** (%)
RoBERTa-Large (w/o KG)	73.1 (±0.5)	68.7 (±0.6)
RGCN (Schlichtkrull et al., 2018)	72.7 (±0.2)	68.4 (±0.7)
GconAttn (Wang et al., 2019)	72.6 (±0.4)	68.6 (±1.0)
KagNet (Lin et al., 2019)	73.5 (±0.2)	69.0 (±0.8)
RN (Santoro et al., 2017)	74.6 (±0.9)	69.1 (±0.2)
MHGRN (Feng et al., 2020)	74.5 (±0.1)	71.1 (±0.8)
QA-GNN (Yasunaga et al., 2021)	76.5 (±0.2)	73.4 (±0.9)
GREASELM (**Ours**)	**78.5** (±0.5)	**74.2** (±0.4)

Table 3: **Test Accuracy comparison on *OpenBookQA***. Experiments are controlled using the same seed LM for all LM+KG methods.

Model	**Acc.**
AristoRoBERTa (no KG)	78.4
+ RGCN	74.6
+ GconAttn	71.8
+ RN	75.4
+ MHGRN	80.6
+ QA-GNN	82.8
GREASELM (**Ours**)	**84.6**

Table 4: **Test accuracy comparison to public *OpenBookQA* model implementations**. *UnifiedQA (11B params) and T5 (3B) are 30x and 8x larger than our model.

Model	**Acc.**	**# Params**
ALBERT (Lan et al., 2020) + KB	81.0	∼235M
HGN (Yan et al., 2020)	81.4	≥355M
AMR-SG (Xu et al., 2021)	81.6	∼361M
ALBERT + KPG (Wang et al., 2020)	81.8	≥235M
QA-GNN (Yasunaga et al., 2021)	82.8	∼360M
T5* (Raffel et al., 2020)	83.2	**∼3B**
T5 + KB	85.4	**≥11B**
UnifiedQA* (Khashabi et al., 2020)	**87.2**	**∼11B**
GREASELM (**Ours**)	84.6	∼359M

Large (Liu et al., 2019) for *CommonsenseQA*, and AristoRoBERTa[1] (Clark et al., 2019) for *OpenBookQA*. For *MedQA-USMLE*, we use a state-of-the-art biomedical language model, SapBERT (Liu et al., 2020), which is an augmentation of PubmedBERT (Gu et al., 2020) that is trained with entity disambiguation objectives to allow the model to better understand entity knowledge.

Existing LM+KG models. We also evaluate GREASELM's ability to exploit its knowledge graph augmentation by comparing with existing LM+KG methods, which incorporate the KGs in different ways compared to GREASELM: (1) Relation Network (RN) (Santoro et al., 2017), (2) RGCN (Schlichtkrull et al., 2018), (3) GconAttn (Wang et al., 2019), (4) KagNet (Lin et al., 2019), (5) MHGRN (Feng et al., 2020), and (6) QA-GNN (Yasunaga et al., 2021). QA-GNN is the existing top-performing model under this LM+KG paradigm. The key difference between GREASELM and these baseline methods is that they do not fuse the representations of both modalities across multiple interaction layers, allowing the representation of both modalities to affect the other (§3.3). For fair comparison, we use the same LM in all the baselines as in our model.

5 EXPERIMENT RESULTS

Our results in Tables 2 and 3 demonstrate a consistent improvement on the CommonsenseQA and OpenbookQA datasets. On CommonsenseQA, our model's test performance improves by 5.5% over fine-tuned LMs and 0.9% over existing LM+KG models. On OpenbookQA, these improvements are magnified, with 6.2% over raw LMs, and 1.8% over the prior best LM+KG system, QA-GNN. The boost over QA-GNN suggests that GREASELM's multi-layer fusion components that pass informa-

[1]OpenBookQA provides an extra corpus of scientific facts in a textual form. AristoRoBERTa is based off RoBERTa-Large, but uses the facts corresponding to each question, prepared by Clark et al. (2019), as an additional input along with the QA context.

Table 5: Performance of GREASELM on the CommonsenseQA IH-dev set on complex questions with semantic nuance such as prepositional phrases, negation terms, and hedge terms. GREASELM consistently outperforms both RoBERTa and QA-GNN in these more complex reasoning settings.

Model	**# Prepositional Phrases**					**Negation Term**	**Hedge Term**
	0	1	2	3	4		
RoBERTa-Large	66.7	72.3	76.3	74.3	69.5	64.6	69.5
QA-GNN	**76.7**	76.2	79.1	74.9	81.4	67.1	74.7
GREASELM (**Ours**)	75.7	**79.3**	**80.4**	**77.2**	**84.7**	**69.5**	**76.2**

tion between the text and KG representations is more expressive than LM+KG methods which do not integrate such sustained interaction between both modalities. We also achieve competitive results to other systems on the leaderboard of OpenBookQA (Table 4), posting the third highest score. However, we note that the T5 (Raffel et al., 2020) and UnifiedQA (Khashabi et al., 2020) models are pretrained models with $8\times$ and $30\times$ more parameters, respectively, than our model. Among models with comparable parameter counts, GREASELM achieves the highest score. Parameter counts were taken from system descriptions on leaderboards or retrieved from published works.

Analysis. Given these overall performance increases, we investigated whether GREASELM's improvements were reflected in questions that required more complex reasoning. Because we had no gold structures from these datasets to categorize the reasoning complexity of different questions, we defined three proxies: the number of prepositional phrases in the questions, the presence of negation terms, and the presence of hedging terms. We use the number of prepositional phrases as a proxy for the number of explicit reasoning constraints being set in the questions. For example, in the CommonsenseQA question in Table 1, "A weasel has a thin body and short legs to easier burrow after prey in a what?" has three prepositional phrases: *to easier burrow*, *after prey*, *in a what*, which each provide an additional constraint in the search for the answer (*n.b.*, in certain cases, the prepositional phrases do not provide constraints that are needed for selecting the correct answer). The presence of negation and hedging terms stratifies our evaluation to questions that have explicit negation mentions (*e.g.*, *no*, *never*) and terms indicating uncertainty (*e.g.*, *sometimes*; *maybe*), respectively.

Our results in Table 5 demonstrate that GREASELM generally outperforms RoBERTa-Large and QA-GNN for both questions with negation terms and hedge terms, indicating GREASELM handles contexts with nuanced constraints. Furthermore, we also note that GREASELM performs better than the baselines across all questions with prepositional phrases, our measure for reasoning complexity. QA-GNN and GREASELM perform comparably on questions with no prepositional phrases, but the increasing complexity of questions requires deeper cross-modal fusion between language and knowledge representations. While QA-GNN's end fusion approach of initializing a node in the GNN from the LM's final representation of the context is an effective approach, it compresses the language context to a single vector before allowing interaction with the KG, potentially limiting the cross-relationships between language and knowledge that can be captured. Interestingly, we note that both GREASELM and QA-GNN significantly outperform RoBERTa-Large even when no prepositional phrases are in the question. We hypothesize that some of these questions may require less reasoning, but require specific commonsense knowledge that RoBERTa may not have learned during pretraining (*e.g.*, 'What is a person considered a bully known for?').

5.1 ABLATION STUDIES

In Table 6, we summarize an ablation study conducted using the CommonsenseQA IHdev set.

Modality interaction. A key component of GREASELM is the connection of the LM to the GNN via the modality interaction module (Eq. 11). If we remove modality interaction, the performance drops significantly, from 78.5% to 76.5% (approximately the performance of QA-GNN). Integrating the modality interaction in every other layer instead of consecutive layers also hurts performance. A possible explanation is that skipping layers could impede learning consistent representations across layers for both the LM and the GNN, a property which may be desirable given we initialize the model using a pretrained LM's weights (*e.g.*, RoBERTa). We also find that sharing parameters

Table 6: **Ablation study** of our model components, using the CommonsenseQA IH-dev set.

Ablation Type	Ablation	Dev Acc.
GREASELM	-	78.5
Modality Interaction	No interaction	76.5
	Interaction in every other layer	76.3
Interaction Layer Parameter Sharing	No parameter sharing	77.1
Graph Connectivity	Context node connected to all nodes in $\mathcal{V}_{\text{sub}}$, not only $\mathcal{V}_{\text{linked}}$	77.6
Node Initialization	Random	60.8
	TransE (Bordes et al., 2013)	77.7

between modality interaction layers (Eq. 11) outperforms not sharing, possibly because our datasets are not very large (*e.g.*, 10k for CommonsenseQA), and sharing parameters helps prevent overfitting.

Graph connectivity. The interaction node e_{int} is a key component of GREASELM that bridges the interaction between the KG and the text. Selecting which nodes in the KG are directly connected to e_{int} affects the rate at which information from different portions of the KG can reach the text representations. We find that connecting e_{int} KG nodes explicitly linked to the input text performs best. Connecting e_{int} to all nodes in the subgraph (*e.g.*, bridge entities) hurts performance (-0.9%).

KG node embedding initialization. Effectively initializing KG node representations is critical. When we initialize nodes randomly instead of using the BERT-based initialization method from Feng et al. (2020), the performance drops significantly (78.5%→60.8%). While using standard KG embeddings (*e.g.*, TransE; Bordes et al., 2013) recovers much of the performance drop (77.7%), we still find that using BERT-based entity embeddings performs best.

5.2 DOMAIN GENERALITY

Our reported results thus far demonstrate the viability of our method in the general commonsense reasoning domain. In this section, we explore whether GREASELM could be adapted to other domains by evaluating on the MedQA-USMLE dataset. Our results in Table 7 demonstrate that GREASELM outperforms state-of-the-art fine-tuned LMs (e.g., SapBERT) and a QA-GNN augmentation of SapBERT. Additionally, we note the improved performance over all classical methods and LM methods first reported in Jin et al. (2021). While these results are promising as they suggest that GREASELM is an effective augmentation of pretrained language models across different domains and KGs (i.e., the biomedical domain with the DDB + Drugbank KG in addition to the commonsense domain), there is still ample room for improvement on this task.

Table 7: Performance on *MedQA-USMLE*

Methods	**Acc.** (%)
Baselines (Jin et al., 2021)	
CHANCE	25.0
PMI	31.1
IR-ES	35.5
IR-CUSTOM	36.1
MAX-OUT	28.6
BERT-BASE-EN	34.3
CLINICALBERT-BASE	32.4
BIOROBERTA-BASE	36.1
BIOBERT-BASE	34.1
ROBERTA-LARGE	35.0
BIOBERT-LARGE	36.7
Baselines (Our implementation)	
SapBERT-Base (w/o KG)	37.2
QA-GNN	38.0
GREASELM (**Ours**)	**38.5**

6 CONCLUSION

In this paper, we introduce GREASELM, a new model that enables interactive fusion through joint information exchange between knowledge from language models and knowledge graphs. Experimental results demonstrate superior performance compared to prior KG+LM and LM-only baselines across standard datasets from multiple domains (commonsense and medical). Our analysis shows improved capability modeling questions exhibiting textual nuances, such as negation and hedging.

7 ETHICS STATEMENT

We outline potential ethical issues with our work below. First, GREASELM is a method to fuse language representations and knowledge graph representations for effective reasoning about textual situations. Consequently, GREASELM could reflect many of the same biases and toxic behaviors exhibited by language models and knowledge graphs that are used to initialize it. For example, prior large-scale language models have been shown to encode biases about race, gender, and other demographic attributes (Sheng et al., 2020). Because GREASELM is seeded with pretrained language models that often learn these patterns, it is possible to reflect them in open-world settings. Second, the ConceptNet knowledge graph (Speer et al., 2017) used in this work has been shown to encode stereotypes (Mehrabi et al., 2021), rather than completely clean commonsense knowledge. If GREASELM were used outside these standard benchmarks in conjunction with ConceptNet as a KG, it might rely on unethical relationships in its knowledge resource to arrive at conclusions. Consequently, while GREASELM could be used for applications outside these standard benchmarks, we would encourage implementers to use the same precautions they would apply to other language models and methods that use noisy knowledge sources.

Another source of ethical concern is the use of the MedQA-USMLE evaluation. While we find clinical reasoning using language models and knowledge graphs to be an interesting testbed for GREASELM and for joint language and reasoning models in general, we do not encourage users to use these models for real world clinical prediction, particularly at these performance levels.

8 REPRODUCIBILITY STATEMENT

To encourage reproducibility, we have outlined the specific datasets evaluated in our work, and provided a list of hyperparameters for each experiment in Appendix Table 8. Code will be shared privately with the reviewers and ACs, and made public upon release of the paper.

REFERENCES

Olivier Bodenreider. The unified medical language system (UMLS): Integrating biomedical terminology. *Nucleic acids research*, 2004.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD*, 2008.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2013.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Çelikyilmaz, and Yejin Choi. Comet: Commonsense transformers for automatic knowledge graph construction. In *Association for Computational Linguistics (ACL)*, 2019.

Antoine Bosselut, Ronan Le Bras, and Yejin Choi. Dynamic neuro-symbolic knowledge graph construction for zero-shot commonsense question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

Peter Clark, Oren Etzioni, Daniel Khashabi, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, Niket Tandon, et al. From 'f' to 'a'on the NY Regents science exams: An overview of the Aristo project. *arXiv preprint arXiv:1909.01958*, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

Yanlin Feng, Xinyue Chen, Bill Yuchen Lin, Peifeng Wang, Jun Yan, and Xiang Ren. Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

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. *arXiv preprint arXiv:2007.15779*, 2020.

Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. In *AAAI*, 2021.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 2021.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. Unifiedqa: Crossing format boundaries with a single qa system. In *Findings of EMNLP*, 2020.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations (ICLR)*, 2020.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. Self-alignment pretraining for biomedical entity representations. *arXiv preprint arXiv:2010.11784*, 2020.

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

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

G. Marcus. Deep learning: A critical appraisal. *ArXiv*, abs/1801.00631, 2018.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *ACL*, 2019.

Ninareh Mehrabi, Pei Zhou, Fred Morstatter, Jay Pujara, Xiang Ren, and A. G. Galstyan. Lawyers are dishonest? quantifying representational harms in commonsense knowledge resources. *ArXiv*, abs/2103.11320, 2021.

Todor Mihaylov and Anette Frank. Knowledgeable reader: Enhancing cloze-style reading comprehension with external commonsense knowledge. In *Association for Computational Linguistics (ACL)*, 2018.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research (JMLR)*, 2020.

Hongyu Ren and Jure Leskovec. Beta embeddings for multi-hop logical reasoning in knowledge graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Hongyu Ren, Weihua Hu, and Jure Leskovec. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *International Conference on Learning Representations (ICLR)*, 2020.

Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, 2018.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. Towards controllable biases in language generation. In *the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)-Findings, long*, 2020.

Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: A Core of Semantic Knowledge. In *16th International Conference on the World Wide Web*, pp. 697–706, 2007.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=rJXMpikCZ`.

Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, September 2014. ISSN 0001-0782. doi: 10.1145/2629489. URL `https://doi.org/10.1145/2629489`.

Peifeng Wang, Nanyun Peng, Pedro Szekely, and Xiang Ren. Connecting the dots: A knowledgeable path generator for commonsense question answering. *arXiv preprint arXiv:2005.00691*, 2020.

Xiaoyan Wang, Pavan Kapanipathi, Ryan Musa, Mo Yu, Kartik Talamadupula, Ibrahim Abdelaziz, Maria Chang, Achille Fokoue, Bassem Makni, Nicholas Mattei, et al. Improving natural language inference using external knowledge in the science questions domain. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, 2018.

Weiwen Xu, Huihui Zhang, Deng Cai, and Wai Lam. Dynamic semantic graph construction and reasoning for explainable multi-hop science question answering. *arXiv preprint arXiv:2105.11776*, 2021.

Jun Yan, Mrigank Raman, Aaron Chan, Tianyu Zhang, Ryan Rossi, Handong Zhao, Sungchul Kim, Nedim Lipka, and Xiang Ren. Learning contextualized knowledge structures for commonsense reasoning. *arXiv preprint arXiv:2010.12873*, 2020.

An Yang, Quan Wang, Jing Liu, Kai Liu, Yajuan Lyu, Hua Wu, Qiaoqiao She, and Sujian Li. Enhancing pre-trained language representations with rich knowledge for machine reading comprehension. In *Association for Computational Linguistics (ACL)*, 2019.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. QA-GNN: Reasoning with language models and knowledge graphs for question answering. *ArXiv*, abs/2104.06378, 2021.

A APPENDIX

Table 8: Hyperparameter settings for models and experiments

Category	Hyperparameter	Dataset		
		CommonsenseQA	**OpenbookQA**	**MedQA-USMLE**
Model architecture	Number of GNN layers M	5	5	3
	Number of Unimodal LM layers N	19	19	9
	Number of attention heads in GNN	2	2	2
	dimension of node embeddings and the messages in GNN	200	200	200
	all MLPs' (except the MInt operators) hidden dimension	200	200	200
	all MLPs' number of hidden layers	1	1	1
	the MInt operators' hidden dimension	400	200	400
Regularization	dropout rate of the embedding layer, GNN layers and fully-connected layers	0.2	0.2	0.2
Optimization	Learning rate of parameters in LM	1.00E-05	1.00E-05	5.00E-05
	Learning rate of parameters not in LM	1.00E-03	1.00E-03	1.00E-03
	Number of epochs in which LM's parameters are kept frozen	4	4	2
	Optimizer	RAdam	RAdam	RAdam
	Learning rate schedule	constant	constant	constant
	batch size	128	128	128
	Number of epochs	30	70	20
	Max gradient norm (gradient clipping)	1.0	1.0	1.0
Data	Max number of nodes	200	200	200
	Max number of tokens	100	100	512