Modularized Transfer Learning with Multiple Knowledge Graphs for Zero-shot Commonsense Reasoning

Anonymous ACL submission

Abstract

Commonsense reasoning systems should be able to generalize to diverse reasoning cases. However, most state-of-the-art approaches depend on expensive data annotations and overfit to a specific benchmark without learning how to perform general semantic reasoning. To overcome these drawbacks, zero-shot QA systems have shown promise as a robust learning scheme by transforming a commonsense knowledge graph (KG) into synthetic QA-form samples for model training. Considering the increasing type of different commonsense KGs, this paper aims to extend the zero-shot transfer learning scenario into multiple-source settings, where different KGs can be utilized synergetically. Towards this goal, we propose to mitigate the loss of knowledge from the interference among the different knowledge sources, by developing a modular variant of the knowledge aggregation as a new zero-shot commonsense reasoning framework. Results on five commonsense reasoning benchmarks demonstrate the efficacy of our framework, improving the performance with multiple KGs.

1 Introduction

The ability to understand natural language through commonsense reasoning is one of the core focuses in the field of natural language processing. To measure and study the different aspects of commonsense reasoning, several datasets are developed, such as SocialIQA (Sap et al., 2019b), CommonsenseQA (Talmor et al., 2018), and PhysicalIQA (Bisk et al., 2020), each requiring different type of commonsense knowledge (*e.g.*, social, taxonomic, causal, declarative, *etc*) to select the correct answer. While large-scale neural systems (Devlin et al., 2018; Yang et al., 2019; Liu et al., 2019b) have shown human-level accuracy on these benchmarks, recent studies (Mitra et al., 2019) also criticize that these models solve individual datasets, rather than learning how to perform general semantic reasoning. To this end, Ma et al. (2021) suggested zero-shot evaluation as a genuine measure for the reasoning capability of the machine.

Inspired by this new metric, in this work, we focus on building unsupervised zero-shot multiple-choice QA systems. That is, we target an arbitrary commonsense reasoning task where conventional approaches (that rely heavily on task-specific supervision) are not applicable to such zero-shot learning scenarios. To learn QA models without expensive annotation efforts, recent works (Ma et al., 2021; Banerjee and Baral, 2020; Malaviya et al., 2020) propose to generate a synthetic QA dataset using a commonsense KG such as `ATOMIC` (Sap et al., 2019a) and `ConceptNet` (Speer et al., 2017). Such an approach mostly focuses only on one specific type of reasoning relations (*e.g.*, if-then relation, or declarative relation), neglecting the fact that real-world QA systems require simultaneously considering different types of reasoning abilities (*e.g.*, declarative and social, or causal and physical reasoning; Ilievski et al., 2021; Chang et al., 2021).

To consider different types of reasoning, this paper extends ideas from the aforementioned zero-shot learning to the *multi-source* case such that it benefits from different types of commonsense knowledge on individual KGs. For example, `ATOMIC` (Sap et al., 2019a) focuses on social commonsense while `ConceptNet` (Speer et al., 2017) contains conceptual knowledge. A practical approach is multi-task learning (MTL; Caruana, 1997; Liu et al., 2019a), which learns a shared encoder for different synthetic QA datasets from multiple KGs. Despite its effectiveness, MTL scheme suffers from interference among different KGs, which results in forgetting previously learned knowledge when trained on new KG which has different kinds of knowledge (Pilault et al., 2021; Pfeiffer et al., 2021; Wang et al., 2021a; Wu et al., 2020).

To address these limitations, we propose a novel, modularized framework that aims to learn multiple

expert models for KGs, then conduct zero-shot fusion to allow collaboration among KGs. For this purpose, we leverage AdapterFusion (Pfeiffer et al., 2021) where multiple tiny modules between Transformer blocks called adapters (Houlsby et al., 2019) can be combined after independent training, thus allowing a continual integration of the adapters without retraining the entire framework. Specifically, we treat the adapters as different KG-specific experts, and combine them using an attention-like fusion module. To improve the fusion of adapters, we suggest a KG-alignment adapter that guides to the apt *expert adapters*. Here, we use KGs in three different synthetic supervision training: (1) KG-specific QA datasets to train the KG-specific expert adapters, (2) a KG classification datasets to train the KG-alignment adapter, and (3) a balanced mixture of KG-specific QA datasets to train the fusion module. Our modularized method alleviates the interference between different KGs, which is the pitfall of MTL from our empirical observation, and thus combines multiple KGs into a synergetic zero-shot framework.

Our contributions are: (1) We suggest a simple, yet effective KG modularization strategy for the use of multiple KGs in commonsense reasoning. (2) We then explore the use of AdapterFusion (Pfeiffer et al., 2021) for better knowledge aggregation based on the KG modularization in zero-shot setting. We believe that such modularized transfer learning is critical to using different knowledge sources synergetically against interference between them. (3) In extensive experiments on various commonsense reasoning benchmarks, our framework achieves significant improvements over baselines using a single KG, even using multiple KGs, which indicates the robustness in genuine commonsense reasoning. We make our code and resulting models available to the community to facilitate future research in this direction.

2 Related Work & Preliminaries

2.1 Zero-shot Commonsense Reasoning

Many researchers have recently focused on building unsupervised models without any benchmark supervisions (*i.e.*, zero-shot learning). In such zero-shot setting, KGs are often used as an external resource for improving model prior (*e.g.*, continually learned from pre-trained language models) (Banerjee and Baral, 2020; Bosselut and Choi, 2019; Ma et al., 2021), especially for commonsense reasoning, as much existing work couples language models with neural/symbolic commonsense KGs.

However, most of existing work are either assuming the existence of the alignment information between tasks and KGs (Banerjee and Baral, 2020) or an integrated KG (Ma et al., 2021). For example, ATOMIC$^{20}_{20}$ (Hwang et al., 2021), a commonsense KG which incorporates tuples from ConceptNet and ATOMIC with new relations and further crowdsourcing, combines multiple KGs into a new integrated KG, but as widely known (Ilievski et al., 2020; Hwang et al., 2021), heterogeneous schema between different KGs may limit triplets that can be integrated.[1] Rather than such symbolic KG integration with the inevitable loss of knowledge, in this work, we explore the neural KG integration leveraging the multiple KGs without additional processing and alignment information between KG and task.

2.2 Transfer Learning with Modular Approaches

The idea of having specialized parameters, or so-called experts, has been widely studied to integrate multiple sources of knowledge via transfer learning. The adapter module (Rebuffi et al., 2017; Houlsby et al., 2019) has been explored as one of such approaches, introducing a small number of task-specific parameters at every layer of pre-trained language model (PLM) while sharing the parameters of underlying PLM which is fixed. To address the limitations of transfer learning due to high re-training cost, many works utilize the multiple adapter modules for individual tasks with different domains (Puigcerver et al., 2020; Bapna et al., 2019; Rücklé et al., 2020; Madotto et al., 2021) considering each adapter to be an expert of each domain. Similar to our work, K-Adapter (Wang et al., 2021a) encodes factual and linguistic knowledge to each adapter, but in this paper, we further explore how to mitigate catastrophic forgetting or interference among multiple adapters for better knowledge transfer in zero-shot setting.

2.3 Multi-Task Learning

MTL (Liu et al., 2019a; Zhang and Yang, 2017; Caruana, 1997) learns a shared representation while aggregating knowledge across multiple learning tasks, often leading to better generalization ability of a model. However, parametric aggregation of

[1] Only 172K tuples of the 3.4M tuples and 5 relations of 36 relations in ConceptNet are integrated into ATOMIC$^{20}_{20}$.

knowledge with MTL has following limitations: (1) retraining the full model when adding new tasks (Houlsby et al., 2019; Pfeiffer et al., 2021, 2020b) (2) catastrophic forgetting and interference between tasks leading to difficulties of solving each task equally well (Pilault et al., 2021; Wu et al., 2020; Yu et al., 2020) and (3) inconsistent effect (Lourie et al., 2021). To deal with these challenges, Mixture-of-Experts (MoE) is a parameterized generalization of ensembling techniques, which has been adapted for MTL with gating network trained to optimize each task (Ma et al., 2018). However, simple linear gating networks are too shallow and thus may destruct task knowledge for commonsense reasoning.

To address this problem, AdapterFusion (Pfeiffer et al., 2021) has been proposed to fuse task specific parameters called adapters for the given target task leveraging attention-like mechanism. AdapterFusion aggregates adapters, which is trained independently for each task, in a non-destructive manner mitigating aforementioned MTL problems such as forgetting and interference between tasks. Recently, it has been used for zero-shot cross-lingual transfer framework (Pfeiffer et al., 2020c; Wang et al., 2021b), which motivates our work to transfer multi-source knowledge with less interference for zero-shot commonsense reasoning.

3 Modularized Zero-shot Framework

In our setup, we repurpose synthetic QA generation (Ma et al., 2021) for the task of knowledge-driven zero-shot learning for commonsense reasoning, *i.e.*, we transform a KG into multiple (Q_i, A_i) pairs where Q_i is a natural language question and $A_i = \{A_{i,1}, ..., A_{i,m}\}$ is the set of options with m answer candidates. Specifically, given a triple (e^{head}, r, e^{tail}) in a KG, where e^{head}, e^{tail} and r denote head/tail entity and relation respectively, we transform e^{head} and r into a natural language question Q_i using templates. For the option set A_i, we use the combination of the correct answer e^{tail} and $m-1$ distractors which are tail entities from other triples sampled randomly (Ma et al., 2021) (details are described in Appendix B).

Formally, we denote (Q_i, A_i) as one QA sample. The goal is to learn a QA model from the synthetic QA sample. In a downstream task (*e.g.*, reasoning benchmarks such as SocialIQA and CommonsenseQA), we need to predict answers given non-synthetic test samples (Q^{test}, A^{test}). In the training stage, we are given K KG-driven datasets $\{\mathcal{D}^k_{QA}\}^K_{k=1}$ from K different KGs, where $\mathcal{D}^k_{QA}$ is a dataset with N_k samples for KG k as follows:

$$\mathcal{D}^k_{QA} = \{(Q_i, A_i, label)\}^{N_k}_{i=1} \quad (1)$$

where $label$ is the index of the correct answer for each sample. In this work, as shown in Table 1, we generate four synthetic QA datasets from ATOMIC, ConceptNet, WikiData, and WordNet (More details are in Appendix C).

QA from ATOMIC (Sap et al., 2019a)
Q: Dana speeds on the highway. Dana is seen as A1: considerate **A2: risky**(✓) A3: lazy
QA from ConceptNet (Speer et al., 2017)
Q: pentode is a [MASK] A1: ascocarp A2: girls footwear **A3: vacuum tube**(✓)
QA from WikiData (Vrandečić and Krötzsch, 2014)
Q: badminton is a [MASK] A1: fable A2: juvenile justice **A3: type of sport**(✓)
QA from WordNet (Miller, 1995)
Q: princewood is part of [MASK] A1: shaddock **A2: genus Cordia**(✓) A3: family Columbidae

Table 1: Synthetic QA examples. We use templates to convert (e^{head}, r) into a natural language sentence.

For effective use of multiple KGs at once with less interference, we present a modularized framework, which is a novel approach to knowledge transfer for the zero-shot setting as shown in Figure 1. As a modular approach, we train the multiple KG-specific adapters (*expert adapters*) with each dataset from KG. Based on these pre-trained adapters, we use a zero-shot fusion method to aggregate knowledge of each adapter leveraging AdapterFusion (Pfeiffer et al., 2021) as a base architecture with the balanced mixture of each KG dataset. Further, for better knowledge fusion, we suggest a KG-alignment aware adapter (*KG-Classifier adapter*) as a guide for detecting alignment with given sample in zero-shot reasoning. Here, we utilize KG classification dataset by verifying the synthetic QA from the KGs. We describe the overall process of our proposed framework in Algorithm 1 (Appendix) and summarize the notations used in this paper in Appendix A.

3.1 KG Modularization

First, we modularize the KGs to preserve their intrinsic knowledge. Considering the importance of using a suitable and well-aligned KG (Ma et al., 2019, 2021) on a downstream task, the subtle difference between each KG should be learned by the model without any interference from each

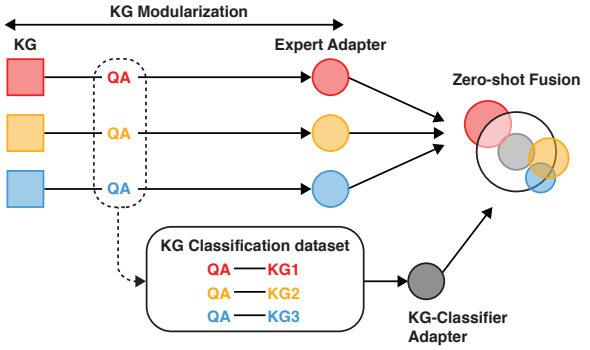


Figure 1: Illustration of the proposed modularized framework for zero-shot commonsense reasoning. Each colored square represents different KGs. Not only for KG modularization, we re-use a set of synthetic QA datasets from the multiple KGs for the purpose of KG classification and zero-shot fusion, which enables better knowledge aggregation.

other. Accordingly, we adopt the adapter module (Houlsby et al., 2019) which repurposes a pre-trained language model (PLM) to incorporate each KG as tiny modules in between Transformer blocks. Specifically, as illustrated in Figure 2 (except for green area), the adapter training strategy involves injecting new layers (parameterized by Φ) into the original PLM (parameterized by θ). The weights of the original PLM are untouched, while the new adapter layers are initialized at random. Formally, we call each adapter trained with $\mathcal{D}^k_{QA}$ as an *expert adapter* for KG k, parameterized by Φ^k_{QA}.

When a QA sample (Q_i, A_i) is given for dataset $\mathcal{D}^k_{QA}$, we first concatenate question Q_i and each answer option $A_i = \{A_{i,1}, ..., A_{i,m}\}$ to generate input sequences $T_i = \{T_{i,1}, ..., T_{i,m}\}$. Then, we compute a score $S_{i,j}$ (Ma et al., 2021) for the answer candidate $A_{i,j}$ is computed as follows:

$$S_{i,j} = -\frac{1}{|T_{i,j}|}\sum_{t=1}^{|T_{i,j}|} logP(w_t|...w_{t-1}, w_{t+1}...; \theta, \Phi) \quad (2)$$

where w_t is a word token in the sequence $T_{i,j}$ and P is the conditional probability from Transformer blocks parameterized by θ and Φ. To train the adapter Φ^k_{QA}, we use the marginal ranking loss (Ma et al., 2021) as follows:

$$\mathcal{L}_{QA} = \frac{1}{m}\sum_{i=1}^{N_k}\sum_{\substack{j=1 \\ j\neq label}}^{m} max(0, \eta - S_{i,label} + S_{i,j}) \quad (3)$$

where η represents the margin.

$$\Phi^k_{QA} \leftarrow \underset{\Phi}{\text{argmin}}\, \mathcal{L}_{QA}(\mathcal{D}^k_{QA}; \theta, \Phi) \quad (4)$$

where KG-invariant parameters θ are fixed and only KG-dependent parameters Φ^k_{QA} are learned, which enables to store the corresponding knowledge separately without any interference. Further, we can parallelize the training of adapter for all KGs. The efficiency of adapter training allows our modularization to be more scalable.

3.2 Zero-shot Fusion

Once the expert adapters are learned, we combine the knowledge from each expert adapter using an attention-like mechanism. We present a novel fusion strategy as shown in Figure 2, which is referred to as the zero-shot fusion. In contrast to AdapterFusion (Pfeiffer et al., 2021) where the focus is learning to transfer knowledge to a specific target task, our zero-shot fusion aims to generalize this transfer to any arbitrary target task. Specifically, the zero-shot fusion parameters Ψ learn to combine fixed expert adapters which are parameterized by θ and Φ^k_{QA}. In each Transformer layer l of PLM with the injected fusion layer, the zero-shot fusion parameters Ψ_{QA} consist of query, key, and value matrices, denoted by $\mathbf{W}^Q_l$, $\mathbf{W}^K_l$, and $\mathbf{W}^V_l$ respectively. These parameters are used to learn the balancing between the representation of each *expert adapters* through attention-like mechanism. While fixing both the parameters θ and all expert adapters $\Phi^1_{QA}, ..., \Phi^K_{QA}$, the only trainable weights Ψ_{QA} on the fusion layer learns to combine the knowledge from different K *expert adapters* by using the subset of $\{\mathcal{D}^k_{QA}\}^K_{k=1}$ by random sampling. Here, we balance the ratio between the K knowledge-driven datasets as N samples (details are in Appendix D). Formally,

$$\Psi_{QA} \leftarrow \underset{\Psi}{\text{argmin}} \sum_{k=1}^{K} \mathcal{L}_{QA}(\mathcal{D}^k_{QA}; \theta, \{\Phi^k_{QA}\}^K_{k=1}, \Psi) \quad (5)$$

where Ψ refers to the initialized zero-shot fusion parameters.

More specifically, in the l-th Transformer layer, let h^l_{PLM} and $h^{k,l}_E$ be the representations of underlying PLM parameterized by θ and an *expert adapter* parameterized by Φ^k_{QA}, respectively. Then, using the hidden representation h^l_{PLM} of PLM as a query, the fusion layer performs the attention-like

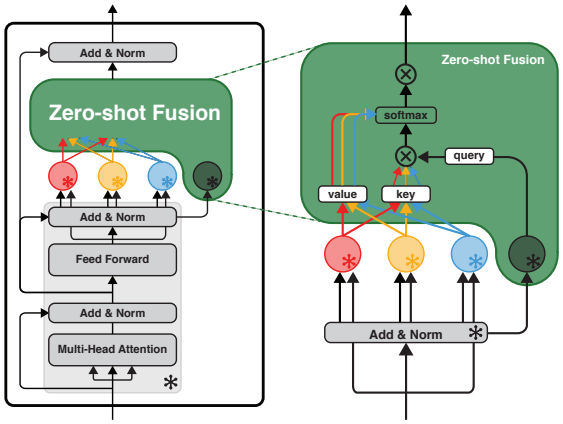


Figure 2: Illustration of the zero-shot fusion architecture with *KG-Classifier adapter*. Each colored circle represents *expert adapters*, except the black circle which denotes *KG-Classifier adapter*. $*$ indicates the fixed layer. Details are in Appendix F

function as follows:

$$\mathbf{K}_l, \mathbf{V}_l = [h_E^{1,l}, ..., h_E^{K,l}] \quad (6)$$

$$\mathbf{Q}_l = h_{PLM}^l \quad (7)$$

$$\mathbf{z}_l = \text{Attention}(\mathbf{Q}_l\mathbf{W}_l^Q, \mathbf{K}_l\mathbf{W}_l^K, \mathbf{V}_l\mathbf{W}_l^V) \quad (8)$$

where $\mathbf{z}_l$ is passed to the next Transformer layer. Given a sample, the zero-shot fusion learns the suitable balancing parameters between the *expert adapters* for zero-shot reasoning. Eventually, it learns to identify generalizability across commonsense reasoning tasks.

3.3 KG-Classifier Adapter

AdapterFusion uses the PLM hidden representation h_{PLM}^l as a query which is learned when training on a specific downstream task. In our zero-shot setting, however, we use a mixture of synthetic QA for fusion training, which is not exactly a training dataset for a downstream task. To compensate for this issue, we present *KG-Classifier adapter*, which is a KG alignment-aware adapter, which is motivated from the fact that the ability to find which KG has an alignment with the given sample can be helpful as a role of providing a guidance for better performance (Ma et al., 2019, 2021).

Specifically, we propose a novel training task for *KG-Classifier adapter*, which requires predicting the KG for the given sample of the task. For that, given $\{\mathcal{D}_{QA}^k\}_{k=1}^K$, we first transform a QA sample (Q_i, A_i) into a new KG classification sample $[Q_i; A_{i,label}]$ where $[;]$ is the concatenation. Then, we obtain a new label $y_i \in \{0,1\}^K$ indicating the corresponding KG source. The samples are in Appendix E. Formally, KG classification dataset $\mathcal{D}_{KGC}$ is defined as:

$$\mathcal{D}_{KGC} = \{([Q_i; A_{i,label}], y_i)\}_{i=1}^M \quad (9)$$

where M is the total size of $\{\mathcal{D}_{QA}^k\}_{k=1}^K$.

Based on $\mathcal{D}_{KGC}$, we learn the *KG-Classifier adapter* parameterized by θ and Φ_{KGC}. First, a classification sample i is encoded into $h_{CLS} \in \mathbb{R}^H$ then scored as $\hat{y}_i \in \mathbb{R}^K$ with a linear layer $W_{KGC} \in \mathbb{R}^{K\times H}$, *i.e.*, $\hat{y}_i = W_{KGC}h_{CLS}$. Once $\hat{y}_i$ is normalized by a softmax layer, the network is trained to minimize the cross-entropy loss $\mathcal{L}_{KGC}$ between the prediction $\hat{y}_i$ and its ground truth y_i:

$$\Phi_{KGC} \leftarrow \underset{\Phi}{\text{argmin}} \sum_{i=1}^{M} \mathcal{L}_{KGC}(y_i, \hat{y}_i; \theta, \Phi) \quad (10)$$

We propose to use the representation of *KG-Classifier adapter* as a query in attention-like mechanism, referred to as the zero-shot fusion with *KG-Classifier adapter*. That is, using the hidden representation h_{KGC}^l of a *KG-Classifier adapter* parameterized by Φ_{KGC} as a query, we substitute $\mathbf{Q}_l$ in Eq. (11) as follows:

$$\mathbf{Q}_l = h_{KGC}^l \quad (11)$$

The overall zero-shot fusion architecture including *KG-Classifier* is illustrated in Figure 2.

4 Experiments

In this section we evaluate the efficacy of our framework on five commonsense reasoning tasks. We denote *KG-Classifier adapter* by *KG-C adapter*.

4.1 Experimental Settings

All our experiments are conducted in a zero-shot setting, in which the models do not have access to the official training data or labels of the benchmark. For the evaluation, we use the validation set of each benchmark. We use accuracy as a metric.

4.1.1 Benchmarks

We evaluate our proposed framework on five question-answering benchmarks for commonsense reasoning: SocialIQA (SIQA) (Sap et al., 2019b), CommonsenseQA (CSQA) (Talmor et al., 2018), Abductive NLI (a-NLI) (Bhagavatula et al., 2019), PhysicalIQA (PIQA) (Bisk et al., 2020), and WinoGrande (WG) (Sakaguchi et al., 2020). Each commonsense benchmark evaluates a specific kind

Model	KG	a-NLI	CSQA	PIQA	SIQA	WG	Avg.
Random	-	50.0	20.0	50.0	33.3	50.0	40.7
Majority	-	50.8	20.9	50.5	33.6	50.4	41.2
GPT2-L	-	56.5	41.4	68.9	44.6	53.2	52.9
RoBERTa-L	-	65.5	45.0	67.6	47.3	57.5	56.6
Self-talk (Shwartz et al., 2020)	-	-	32.4	70.2	46.2	54.7	-
COMET-DynaGen (Bosselut and Choi, 2019)	AT	-	-	-	50.1	-	-
SMLM (Banerjee and Baral, 2020)	*	65.3	38.8	-	48.5	-	-
RoBERTa-L (MR) (Ma et al., 2021)	AT	70.8	64.2	72.1	63.1	59.2	65.9
RoBERTa-L (MR) (Ma et al., 2021)	CN,WD,WN	70.0	67.9	72.0	54.8	59.4	64.8
RoBERTa-L (MR) (Ma et al., 2021)	Whole	70.5	67.4	72.4	63.2	**60.9**	66.9
MTL	Whole	69.8 (± 0.5)	66.0 (± 0.9)	71.2 (± 0.8)	62.2 (± 1.0)	59.5 (± 0.2)	65.7
zero-shot fusion w/o *KG-C adapter*	Whole	72.3(±0.4)	67.9(±0.2)	**73.1**(±0.4)	65.9(±0.5)	59.7(±0.2)	67.8
zero-shot fusion w/ *KG-C adapter*	Whole	**72.5**(±0.2)	**68.2**(±0.2)	72.9(±0.4)	**66.6**(±0.1)	60.8(±0.1)	**68.2**

Table 2: Zero-shot evaluation results with different combinations of models and knowledge sources across five commonsense tasks. AT, CN, WD and WN represent ATOMIC, ConceptNet, WikiData and WordNet, respectively. Whole represents the combination of AT, CN, WD and WN. Bold text indicates the best performance on each benchmark. RoBERTa-L (MR) used the synthetic dataset after filtering, while we use the raw version. SMLM (*) used different KG which has strong alignment with each task (*e.g.*AT for SIQA).

of knowledge: social commonsense for SIQA, concept-level commonsense for CSQA, abductive reasoning for a-NLI, physical commonsense for PIQA, and pronoun resolution ability for WG.[2] The details are presented in Appendix G.

4.1.2 Baselines

We compare our framework with the following baselines. First, to show the characteristics of each benchmark, we use the random or the most frequent label as *Random* and *Majority* baseline, respectively. RoBERTa-L and GPT2-L is the performance of each PLM without any fine-tuning. Also, as the baseline for the unsupervised learning model using KGs, we report the performance of Self-talk, COMET-DynaGen, SMLM as presented in original papers.

For further analysis in §4.4 and §4.5, we set the following models that are pre-trained on the synthetic QA datasets from KGs as baselines:

- **Single-Task Learning (STL)**: The model is pre-trained on a synthetic QA dataset generated from a single KG. Specifically, we experiment two architectural choices: PLM (STL-PLM) and PLM with adapters (STL-Adapter). For each architecture, there are four STL models for each of synthetic QA datasets derived from ATOMIC, ConceptNet, WikiData, and WordNet. We note that the trained STL-Adapter is an *expert adapter* from a specific KG in our framework.
- **Multi-Task Learning (MTL)**: The model is pre-trained on multiple synthetic QA datasets, each of which is generated from a KG. We experiment with a PLM trained on all four aforementioned synthetic QA datasets. We note that the difference between STL-PLM and MTL is whether to use one synthetic QA dataset or multiple synthetic QA datasets for its training.

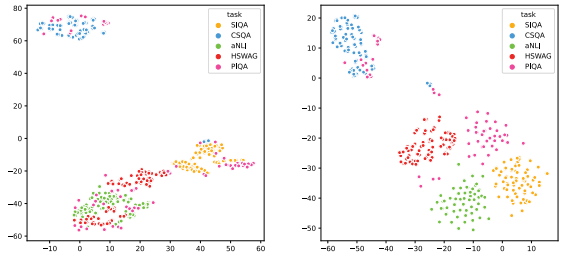
4.1.3 Implementations

We employ RoBERTa-L (Liu et al., 2019b) from Hugging Face's transformers toolkit for all experiments. We follow the default settings from Ma et al. (2021). Our implementation uses Adapter (Houlsby et al., 2019) and AdapterFusion (Pfeiffer et al., 2021) as a base model architecture from AdpaterHub (Pfeiffer et al., 2020a). We run our experiments with three different random seeds. We describe the implementation details in the Appendix H.

4.2 Main Results

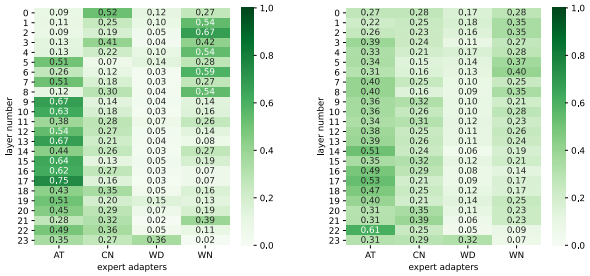
Table 2 shows the zero-shot evaluation results on five benchmark datasets. Generally, zero-shot fusion scores higher than the baselines across all benchmarks, and further, zero-shot fusion shows the best performance in all benchmarks except WG. We note that although Ma et al. (2021) uses the synthetic QA dataset after sample filtering, our method achieves comparable performance with the best performance in WG, even with the raw dataset. Also, the average score of all evaluation benchmarks (the last column of Table 2) shows that zero-shot fusion has generalisability in commonsense reasoning.

[2]Some benchmarks have a strong alignment with a certain KG due to its construction strategy: SIQA-ATOMIC, and CSQA-ConceptNet. To make a direct comparison with Ma et al. (2021), we use the same KGs to generate data samples.



(a) PLM (b) *KG-Classifier adapter*

Figure 3: t-SNE visualization of the hidden representation from (a) PLM and (b) *KG-C adapter*. Each color denotes the five different benchmark samples.



(a) w/o *KG-C adapter* (b) w/*KG-C adapter*

Figure 4: Comparison of attention probability between zero-shot fusion with/without *KG-C adapter*. The x- and y-axis indicate *expert adapters* and the fusion layer number in RoBERTa-L, respectively. The darker color indicates higher attention probability in fusion layer.

In addition, zero-shot fusion achieves consistent improvements over MTL. These results indicate that our proposed zero-shot fusion method attributes to fusing the knowledge of multiple KGs more synergetically regardless of the task.

Moreover, as an ablation, we compare the zero-shot fusion with and without *KG-C adapter* to explore the efficacy of the *KG-C adapter*. We can observe that zero-shot fusion with *KG-C adapter* improves the average accuracy by 0.4%, which implies that the use of *KG-C adapter* improves the overall performance and makes our method generalize better on most of the evaluation benchmarks.

4.3 Impact of the KG-Classifier Adapter

To assess the effects of the *KG-C adapter* itself, we visualize and compare the final layer `[CLS]` token representation between PLM and *KG-C adapter*. Figure 3 shows t-SNE (Van der Maaten and Hinton, 2008) plots of all representation of five benchmark datasets. In this figure, every sample is mapped into a 1024-dimensional feature space through RoBERTa-L model and projected back into a two-dimensional plane by t-SNE. We can observe that *KG-C adapter* can separate the samples of different benchmarks well despite being unseen data. It verifies that KG-awareness acquired with the KG classification task is beneficial to categorize the given sample. The *KG-C adapter* can thus generate a relevant KG-aware query for a given sample and help to fuse representations from suitable *expert adapters* in our proposed framework.

Further, we explore how the *KG-C adapter* affects zero-shot fusion which is based on an attention-like mechanism (Pfeiffer et al., 2021) compared to zero-shot fusion without *KG-C adapter*. Here, while zero-shot fusion without *KG-C adapter* simply uses the representation of PLM as a query, zero-shot fusion with *KG-C adapter* leverages the representation of *KG-C adapter*. To illustrate this strength, we visualize the attention probability of `[CLS]` token from each fusion layer as a representative in Figure 4. The column of the darker cell indicates the adapter that has the bigger influence on the fused representation. We can observe that zero-shot fusion with *KG-C adapter* fuses the knowledge from different experts with a subtle difference rather than focusing on a single expert severely. This implies that *KG-C adapter* enables the delicate balancing between multiple knowledge sources based on the KG-alignment awareness, which leads to performance improvements in commonsense reasoning tasks. Interestingly, both cases have the ability not to focus on the *expert adapter* based on `WikiData`, which can be seen as a redundant expert.[3] This observation would benefit from the further study that explores the optimal combination of KGs by expert selection or rejection.

4.4 Mitigating Interference

In this experiment, we compare the amount of interference in the MTL and zero-shot fusion with *KG-C adapter*. We propose a novel evaluation metric, the *interference ratio*, which is the percentage of the incorrectly predicted samples by the multi-KG models among the correctly predicted samples from the STL models in common.

Using the interference ratio, we can precisely compare the negative effects of multi-KG models on knowledge aggregation since the only reason to get the correct samples wrong is the interfer-

[3]The zero-shot fusion with KG-C adapter using AT, CN, and WN shows the best average performance in Table 10.

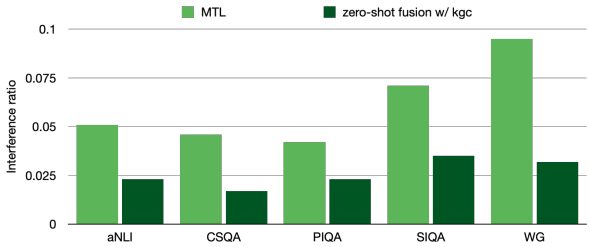
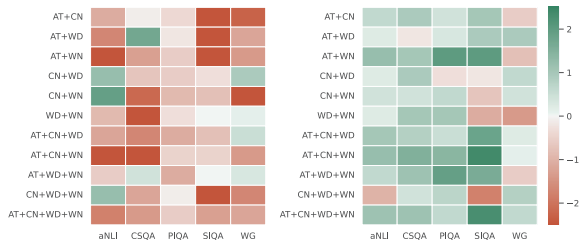


Figure 5: Interference ratio of multi-KG models on five benchmarks. The lower indicates less interference.



(a) MTL

(b) zero-shot fusion w/ *KG-C adapter*

Figure 6: Relative improvement upon the STL on five benchmarks. The x- and y-axis indicate the benchmark and the combination of the KGs, respectively. The value of each cell indicates the relative performance improvement of using multiple KGs over the highest performance among STLs. The green and red colors denote the improvement or decrease of relative performance, respectively.

ence caused by learning with additional KGs. We present the interference ratio of the models on five benchmark datasets in Figure 5. This figure shows that MTL has the higher interference ratio than the competing models across all benchmarks. Our method achieves a substantially better ratio, especially when *KG-C adapter* is used. This demonstrates the efficacy of our framework in mitigating interference between knowledge, which is one of the major problems of MTL.

4.5 Visualization of Knowledge Aggregation

To verify the ability of our model to aggregate different types of KGs, we compare the relative performance gains of MTL and zero-shot fusion with *KG-C adapter* when increasing the number of KGs. The performance of all KG-combinations for each framework is presented in Table 9 and Table 10. We visualize the improvement of performance for five benchmark development sets, leveraging heatmaps in Figure 6. Here, for the sake of brevity, we denote our framework with *KG-C adapter* as our method.

For MTL in Figure 6 (a), the color of the cell denotes the relative improvement of MTL with the combination of KGs over the best performance among the STL-PLM of KGs. Also, for our method in Figure 6 (b), the relative improvement is measured based on the best performance among the STL-Adapter of KGs, considering the difference of the base architecture for MTL (i.e. PLM) and zero-shot fusion (i.e. PLM with adapter). The green and red colors denote the increase and decrease of performance, respectively, when using multiple KGs together. The greener color on the cells indicates that the approach benefits from an increasing number of KGs, which implies aggregating knowledge successfully.

In Figure 6, while the MTL tends to show the decrease of the performance when more KGs are utilized for training, our method obtains relative performance improvement across most of benchmarks. In both framework, the slightly degraded performance of the combination of KGs without ATOMIC could be due to the strong alignment between ATOMIC and SIQA. Except for the above case, we can observe that as more KGs are leveraged, the color of the cell gets greener, which implies that our method gains more advantages for better performance. This demonstrates that our method enables knowledge aggregation for multiple KGs synergetically.

5 Conclusion

Despite the existence of various types of commonsense KGs, utilizing multiple KGs has not been explored enough in the commonsense reasoning field. Motivated by this, this paper proposes a modularized transfer learning framework to fuse the knowledge from multiple KGs efficiently for zero-shot commonsense reasoning. Our framework consists of KG modularization for *expert adapter*, zero-shot fusion and *KG-Classifier adapter*. Extensive experiments show that our framework obtains strong improvements over MTL on five commonsense reasoning benchmarks.

In the future, our work can be extended to adapt our methods to further various multiple KGs with studies of appropriate scale for KG modularization. In addition, based on our hypothesis that the existence of an optimal combination, we can explore the study for the optional use of modularized KG experts for the best transfer learning.

References

Pratyay Banerjee and Chitta Baral. 2020. Self-supervised knowledge triplet learning for zero-shot question answering. *EMNLP*.

Ankur Bapna, Naveen Arivazhagan, and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. *EMNLP*.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Scott Wen-tau Yih, and Yejin Choi. 2019. Abductive commonsense reasoning. *ICLR*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *AAAI*.

Antoine Bosselut and Yejin Choi. 2019. Dynamic knowledge graph construction for zero-shot commonsense question answering. *arXiv e-prints*, pages arXiv–1911.

Rich Caruana. 1997. Multitask learning. *Machine learning*, 28(1):41–75.

Ting-Yun Chang, Yang Liu, Karthik Gopalakrishnan, Behnam Hedayatnia, Pei Zhou, and Dilek Hakkani-Tur. 2021. Incorporating commonsense knowledge graph in pretrained models for social commonsense tasks. *EMNLP-Workshop*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *PMLR*, pages 2790–2799. PMLR.

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

Filip Ilievski, Alessandro Oltramari, Kaixin Ma, Bin Zhang, Deborah L McGuinness, and Pedro Szekely. 2021. Dimensions of commonsense knowledge. *Knowledge-Based Systems*.

Filip Ilievski, Pedro Szekely, Jingwei Cheng, Fu Zhang, and Ehsan Qasemi. 2020. Consolidating commonsense knowledge. *arXiv preprint arXiv:2006.06114*.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Multi-task deep neural networks for natural language understanding. *ACL*.

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

Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Unicorn on rainbow: A universal commonsense reasoning model on a new multitask benchmark. *arXiv preprint arXiv:2103.13009*.

Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *ACM SIGKDD*.

Kaixin Ma, Jonathan Francis, Quanyang Lu, Eric Nyberg, and Alessandro Oltramari. 2019. Towards generalizable neuro-symbolic systems for commonsense question answering. *EMNLP*.

Kaixin Ma, Filip Ilievski, Jonathan Francis, Yonatan Bisk, Eric Nyberg, and Alessandro Oltramari. 2021. Knowledge-driven data construction for zero-shot evaluation in commonsense question answering.

Andrea Madotto, Zhaojiang Lin, Yejin Bang, and Pascale Fung. 2021. The adapter-bot: All-in-one controllable conversational model.

Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. 2020. Commonsense knowledge base completion with structural and semantic context. In *AAAI*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*.

Arindam Mitra, Pratyay Banerjee, Kuntal Kumar Pal, Swaroop Mishra, and Chitta Baral. 2019. Exploring ways to incorporate additional knowledge to improve natural language commonsense question answering. *arXiv preprint arXiv:1909.08855*.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. Adapterfusion: Non-destructive task composition for transfer learning. *EACL*.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020a. Adapterhub: A framework for adapting transformers.

Jonas Pfeiffer, Edwin Simpson, and Iryna Gurevych. 2020b. Low resource multi-task sequence tagging–revisiting dynamic conditional random fields. *arXiv preprint arXiv:2005.00250*.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020c. Mad-x: An adapter-based framework for multi-task cross-lingual transfer. *arXiv preprint arXiv:2005.00052*.

Jonathan Pilault, Amine Elhattami, and Christopher Pal. 2021. Conditionally adaptive multi-task learning: Improving transfer learning in nlp using fewer parameters & less data. *ICLR*.

Joan Puigcerver, Carlos Riquelme, Basil Mustafa, Cedric Renggli, André Susano Pinto, Sylvain Gelly, Daniel Keysers, and Neil Houlsby. 2020. Scalable transfer learning with expert models. *arXiv preprint arXiv:2009.13239*.

Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. 2017. Learning multiple visual domains with residual adapters. *arXiv preprint arXiv:1705.08045*.

Andreas Rücklé, Jonas Pfeiffer, and Iryna Gurevych. 2020. Multicqa: Zero-shot transfer of self-supervised text matching models on a massive scale. *EMNLP*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *AAAI*.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019a. Atomic: An atlas of machine commonsense for if-then reasoning. In *AAAI*.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019b. Socialiqa: Commonsense reasoning about social interactions. *EMNLP*.

Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Unsupervised commonsense question answering with self-talk. *EMNLP*.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *NAACL*.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *JMLR*.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *ACM*.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. 2021a. K-adapter: Infusing knowledge into pre-trained models with adapters. *ACL*.

Xinyi Wang, Yulia Tsvetkov, Sebastian Ruder, and Graham Neubig. 2021b. Efficient test time adapter ensembling for low-resource language varieties.

Sen Wu, Hongyang R Zhang, and Christopher Ré. 2020. Understanding and improving information transfer in multi-task learning. *ICLR*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *NIPS*.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. *NIPS*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *EMNLP*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *ACL*.

Yu Zhang and Qiang Yang. 2017. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*.

A List of Notations

We summarize the notations used in this paper in Table 3.

B Synthetic QA

We generate QA for four KGs (`ATOMIC`, `ConceptNet`, `WikiData` and `WordNet`) based on synthetic QA generation (Ma et al., 2021) without sample filtering. Table 4 shows the statistics of the synthetic QA dataset from KGs. We use the prefixes for relation of triplet as shown in Table 5 for generating synthetic QA. The samples of synthetic QA with source triplet are shown in Table 6.

C Commonsense Knowledge Graphs

A variety of KGs have been proposed to provide large-scale high quality collection of different commonsense knowledge types: `ATOMIC` (Sap et al., 2019a) focuses on inferential knowledge organized as typed if-then relations with variables (*e.g.*, "*if X pays Y a compliment, then Y will likely return the compliment*"). `ConceptNet` (Speer et al., 2017) mainly consists of taxonomic and lexical knowledge (*e.g.*, RelatedTo, Synonym, and IsA) and physical commonsense knowledge (*e.g.*, MadeOf and PartOf). `WikiData` (Vrandečić and Krötzsch, 2014) is a general-domain KG which has a close relation with Wikipedia. `WordNet` (Miller, 1995) is a large lexical source of words and taxonomical system.

D Dataset for Zero-shot Fusion

For zero-shot fusion training, we use balanced mixture of synthetic QA from different KGs by random sampling. The statistics of dataset for zero-shot fusion is shown in Table 7. For validation dataset, we balance between the `ATOMIC`, `ConceptNet` and `WordNet` due to the lack of synthetic QA validation dataset from `WikiData`.

Notation	Meaning
(e^{head}, r, e^{tail})	Triple of KG (head entity, relation, tail entity)
Q_i	Natural language Question of sample i
$A_i = \{A_{i,1}, ..., A_{i,m}\}$	A set of answer options of sample i, $A_{i,j}$ denotes j-th answer option of sample $i (1 \leq j \leq m)$
$T_i = \{T_{i,1}, ..., T_{i,m}\}$	Input sequences generated by concatenation of Q_i and A_i
w_t	A word t-th token in the sequence $T_{i,j}$
$label$	the index of the correct answer for sample
$\mathcal{D}^k_{QA}$	Synthetic QA generated by KG k, $1 \leq k \leq K$
N_k	The number of samples for $\mathcal{D}^k_{QA}$, $1 \leq k \leq K$
θ	Parameters for pre-trained LM
Φ^k_{QA}	Parameters for the *expert adapter* of KG k, $1 \leq k \leq K$
Φ_{KGC}	Parameters for the *KG-Classifier adapter*
Ψ_{QA}	Parameters for the fusion layer
l	The index of Transformer layer
$\mathbf{W}^Q_l$	Query matrix of fusion layer in lth Transformer layer
$\mathbf{W}^K_l$	Key matrix of fusion layer in lth Transformer layer
$\mathbf{W}^V_l$	Value matrix of fusion layer in lth Transformer layer
h^l_{PLM}	Hidden representation of PLM parameterized by θ in lth Transformer layer
$h^{k,l}_E$	Hidden representation of *expert adapter* parameterized by Φ^k_{QA} in lth Transformer layer
h^l_{KGC}	Hidden representation of *KG-Classifier adapter* parameterized by Φ_{KGC} in lth Transformer layer

Table 3: Notations and their meanings

KG	Train	Validation	Total
ATOMIC	534,833	60,289	595,122
ConceptNet	363,645	19,140	382,785
WikiData	42,342	2,229	44,571
WordNet	256,922	13,523	270,445
Whole	1,197,742	95,181	1,292,923

Table 4: Synthetic QA dataset statistics. Whole represents the combination of AT,CN,WD and WN.

relation	prefix
xAttr	. PersonX is seen as
xIntent	. Before, PersonX wanted
xNeed	. Before, PersonX needed to
xReact	. As a result, PersonX felt
xWant	. As a result, PersonX wanted to
xEffect	. PersonX then
oReact	. As a result, others felt
oWant	. As a result, others wanted to
oEffect	. Others then
Causes	can cause [MASK]
UsedFor	can be used for [MASK]
CapableOf	is capable of [MASK]
CausesDesire	causes desire for [MASK]
IsA.	is a [MASK]
SymbolOf	is a symbol of [MASK]
MadeOf	can be made of [MASK]
LocatedNear	is often located near [MASK]
Desires	desires [MASK]
AtLocation	can be found at [MASK]
HasProperty	has property [MASK]
PartOf	is part of [MASK]
HasFirstSubevent	starts by [MASK]
HasLastSubevent	ends by [MASK]

Table 5: Prefixes used for synthetic QA dataset

QA from ATOMIC (Sap et al., 2019a)
(e^h, r, e^t): (Dana speeds on the highway., xAttr, risky)
Q: Dana speeds on the highway. Dana is seen as
A1: considerate **A2: risky**(✓) A3: lazy
QA from ConceptNet (Speer et al., 2017)
(e^h, r, e^t): (pentode, IsA, vacuum tube)
Q: pentode is a [MASK]
A1: ascocarp A2: girls footwear **A3: vacuum tube**(✓)
QA from WikiData (Vrandečić and Krötzsch, 2014)
(e^h, r, e^t): (badminton, IsA, type of sport)
Q: badminton is a [MASK]
A1: fable A2: juvenile justice **A3: type of sport**(✓)
QA from WordNet (Miller, 1995)
(e^h, r, e^t): (princewood, PartOf, genus Cordia)
Q: princewood is part of [MASK]
A1: shaddock **A2: genus Cordia**(✓) A3: family Columbidae

Table 6: Synthetic QA examples. We use templates to convert a question (e^{head}, r) into a natural language.

KG	Train	Validation	Total
+ATOMIC	2,500	2,500	5,000
+ConceptNet	2,500	2,500	5,000
+WikiData	2,500	2,229	4,729
+WordNet	2,500	2,500	5,000
Total	10,000	9,729	19,729

Table 7: Statistics of the dataset for zero-shot fusion

E KG-Classification Dataset

We suggest KG-Classification dataset $\mathcal{D}_{KGC}$ for *KG-Classifier adapter* training. The example of transformation from synthetic QA dataset $\mathcal{D}_{QA}$ is shown in Table 8. The dataset size is equal to the whole dataset of synthetic QA (refer to Table 4).

QA → KG-Classification `ATOMIC`
Q: Dana speeds on the highway. Dana is seen as A1: considerate **A2: risky**(✓) A3: lazy
S: Dana speeds on the highway. Dana is seen as risky. A: Atomic
QA → KG-Classification `ConceptNet`
Q: pentode is a [MASK] A1: ascocarp A2: girls footwear **A3: vacuum tube**(✓)
S: pentode is a vacuum tube. A: ConceptNet
QA → KG-Classification `WikiData`
Q: badminton is a [MASK] A1: fable A2: juvenile justice **A3: type of sport**(✓)
S: badminton is a type of sport. A: WikiData
QA → KG-Classification `WordNet`
Q: princewood is part of [MASK] A1: shaddock **A2: genus Cordia**(✓) A3: family Columbidae
S: princewood is part of genus Cordia. A: WordNet

Table 8: KG-Classification examples from synthetic QA dataset of each KG

F Zero-shot architecture with parameters

We describe the illustration of the zero-shot fusion architecture with parameters in Figure 7.

G Commonsense Reasoning Benchmarks

SocialIQA (SIQA) (Sap et al., 2019b) requires reasoning for emotional and social intelligence in everyday situations. Each QA consists of a context that comes from `ATOMIC`, a question which is based on the relations in `ATOMIC`, and 3 answer candidates. It contains 38,000 multiple-choice questions, which is generated by crowdsourcing.
CommonsenseQA (CSQA) (Talmor et al., 2018) evaluates a broad range of concept-level commonsense reasoning. Each multiple-choice question, answer and distractors are designed by crowdsourcing based on the `ConceptNet`.
Abductive NLI (a-NLI) (Bhagavatula et al., 2019) asks to infer the most plausible explanation based on the given causal situation to test abductive reasoning in narratives. Each sample consists of the beginning and the end of the story with two possible options to be an explanation for the given situation.
PhysicalIQA (PIQA) (Bisk et al., 2020) requires physical commonsense reasoning to select the most sensible solution for the given goal among the two choices. Its dataset is comprised of over 16,000 training samples, 2K validation samples, and 3K test samples.
HellaSWAG (HSWAG) (Zellers et al., 2019) is an evolved version of SWAG (Zellers et al., 2018), which asks to infer the most proper story based on the given situation. The dataset consists of 70K questions with four answer options.

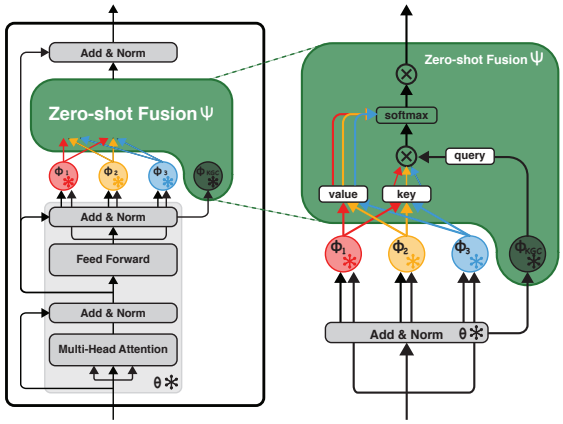


Figure 7: Illustration of the zero-shot fusion architecture with *KG-Classifier adapter*. Each colored circle represents *expert adapters*, , except the black circle which denotes *KG-Classifier adapter*. $*$ indicates the fixed layer.

H Implementation Details

In all our experiments, we use max sequence length 128, batch size 32, weight decay 0.01, adam β_1 0.9, adam β_2 0.99, adam epsilion $1e^{-8}$, warm-up proportion 0.05, and margin 1.0. The experiments are conducted split across NVIDIA GeForce 3090 and NVIDIA RTX A5000.

H.1 Baselines

The baseline models for STL-PLM and MTL are trained with learning rate $1e^{-5}$ for single epoch.

H.2 Adapter

For *expert adapters*, we set the batch size 32, and use learning rate $8e^{-5}$ after tuning in

$\{5e^{-6}, 8e^{-6}, 1e^{-5}, 5e^{-5}, 8e^{-5}, 1e^{-4}\}$. For *KG-Classifier adapter*, we use learning rate $1e^{-5}$, batch size 64 for five epochs.

H.3 Zero-shot fusion

After experiment with learning rates $\{1e^{-5}, 8e^{-5}\}$, we empirically find that a learning rate of $1e^{-5}$ works well on zero-shot fusion without/with *KG-Classifier adapter*, respectively. Here, we set the attention drop probability 0.1. As we used extremely smaller subset of the synthetic QA dataset, zero-shot fusions are trained for five epochs.

I Knowledge aggregation of zero-shot fusion

In order to validate the efficacy on knowledge aggregation of zero-shot fusion over the STL, we present the results of each framework with various combination of KGs in Table 9 and Table 10.

Model	KG	a-NLI	CSQA	PIQA	SIQA	WG	Avg.
STL-PLM	AT	71.6	64.0	72.2	63.2	60.5	66.3
	CN	67.9	68.5	72.6	54.6	58.6	64.4
	WD	68.4	64.7	72.0	53.7	58.6	63.5
	WN	67.2	61.4	71.7	53.5	58.9	62.5
MTL	AT, CN	70.5	68.4	72.2	60.1	58.2	65.9
	AT, WD	69.9	66.4	72.0	60.1	59.3	65.5
	AT, WN	69.1	62.7	71.6	59.1	59.1	64.3
	CN, WD	69.6	67.8	72.0	54.3	59.5	64.6
	CN, WN	69.8	66.3	71.7	53.8	56.4	63.6
	WD, WN	67.5	62.0	71.7	53.7	59.0	62.8
MTL	AT, CN, WD	70.4	66.8	71.5	62.4	61.0	66.4
	AT, CN, WN	68.5	65.7	72.1	62.7	59.1	65.6
	AT, WD, WN	71.0	65.1	71.1	63.2	60.8	66.2
	CN, WD, WN	69.6	67.3	72.5	52.0	57.2	63.7
MTL	AT, CN, WD, WN	69.8	67.1	72.0	61.9	59.3	66.0

Table 9: STL-PLM and MTL performance across five commonsense tasks in various combination of KGs. AT, CN, WD and WN represent `ATOMIC`, `ConceptNet`, `WikiData` and `WordNet`, respectively. We run our experiment with seed 42.

Model	KG	a-NLI	CSQA	PIQA	SIQA	WG	Avg.
STL-Adapter	AT	71.3	66.5	71.1	64.4	60.3	66.7
	CN	70.6	67.2	72.4	55.5	58.7	64.9
	WD	66.8	61.6	69.9	51.8	58.5	61.7
	WN	67.6	60.0	70.3	54.0	57.0	61.8
zero-shot fusion w/*KGC-adapter*	AT, CN	71.9	68.1	72.8	65.4	59.7	67.6
	AT, WD	71.5	66.3	71.4	65.3	61.2	67.1
	AT, WN	72.5	67.5	73.1	66.4	59.5	67.8
	CN, WD	70.8	68.1	72.1	55.3	59.3	65.1
	CN, WN	71.0	67.6	73.0	54.8	59.1	65.1
	WD, WN	67.8	62.6	71.3	52.9	57.1	62.3
zero-shot fusion w/*KGC-adapter*	AT, CN, WD	72.3	68.0	72.9	66.2	60.5	68.0
	AT, CN, WN	72.5	68.7	73.8	66.8	60.4	68.4
	AT, WD, WN	71.9	67.6	73.0	66.0	59.7	67.6
	CN, WD, WN	69.6	67.6	73.1	53.7	59.5	64.7
zero-shot fusion w/*KGC-adapter*	AT, CN, WD, WN	72.4	68.3	73.0	66.7	60.9	68.3

Table 10: STL-Adapter and zero-shot fusion w/ *KG-C adapter* performance across five commonsense tasks in various combination of KGs. AT, CN, WD and WN represent `ATOMIC`, `ConceptNet`, `WikiData` and `WordNet`, respectively. Whole represents the combination of AT, CN, WD and WN. We run our experiment with seed 42.

Algorithm 1: Proposed framework for zero-shot commonsense reasoning

Input: PLM parameters θ, K KGs
Output: Reasoning model parameters $(\theta, \{\Phi_{QA}^k\}_{k=1}^K, \Phi_{KGC}, \Psi_{QA})$
$\{\mathcal{D}_{QA}^k\}_{k=1}^K \leftarrow$ Generate synthetic QA samples from multiple KGs (Eq. 1)
$\mathcal{D}_{KGC} \leftarrow$ Generate KG classification samples from multiple KGs (Eq. 9)
for *each KG* $k = 1, ..., K$ **do**
  $\Phi_{QA}^k \leftarrow \text{argmin}_\Phi \mathcal{L}_{QA}(\mathcal{D}_{QA}^k; \theta, \Phi)$ (Eq. 4)
$\Phi_{KGC} \leftarrow \text{argmin}_\Phi \sum_{i=1}^M \mathcal{L}_{KGC}(\mathcal{D}_{KGC}; \theta, \Phi)$ (Eq. 10)
$\Psi_{QA} \leftarrow \text{argmin}_\Psi \sum_{k=1}^K \mathcal{L}_{QA}(\mathcal{D}_{QA}^k; \theta, \{\Phi_{QA}^k\}_{k=1}^K, \Phi_{KGC}, \Psi)$ (Eq. 5 and 11)
return $(\theta, \{\Phi_{QA}^k\}_{k=1}^K, \Phi_{KGC}, \Psi_{QA})$