COSMO: Conditional SEQ2SEQ-based Mixture Model for Zero-Shot Commonsense Question Answering

Farhad Moghimifar[1] and **Lizhen Qu**[2] and **Yue Zhuo**[3]
Mahsa Baktashmotlagh[1] and **Gholamreza Haffari**[2]
[1]The School of ITEE, The University of Queensland, Australia
[2]Faculty of Information Technology, Monash University, Australia
[3]School of CSE, The University of New South Wales, Australia
{f.moghimifar,m.baktashmotlagh}@uq.edu.au
firstname.lastname@monash.edu, terry.zhuo@unsw.edu.au

Abstract

Commonsense reasoning refers to the ability of evaluating a social situation and acting accordingly. Identification of the implicit causes and effects of a social context is the driving capability which can enable machines to perform commonsense reasoning. The dynamic world of social interactions requires context-dependent on-demand systems to infer such underlying information. However, current approaches in this realm lack the ability to perform commonsense reasoning upon facing an unseen situation, mostly due to incapability of identifying a diverse range of implicit social relations. Hence they fail to estimate the correct reasoning path. In this paper, we present Conditional SEQ2SEQ-based Mixture model (COSMO), which provides us with the capabilities of dynamic and diverse content generation. We use COSMO to generate context-dependent clauses, which form a dynamic Knowledge Graph (KG) on-the-fly for commonsense reasoning. To show the adaptability of our model to context-dependant knowledge generation, we address the task of zero-shot commonsense question answering. The empirical results indicate an improvement of up to +5.2% over the state-of-the-art models.

1 Introduction

People understand narratives of everyday life by capitalising on their commonsense knowledge. They can easily reason about unobserved causes and effects in relation to the events described in narratives, as well as plausible characteristics and mental states of the involved persons. Although this kind of reasoning seems trivial for humans, it is still out of reach for current natural language understanding (NLU) systems. Recently, there have been fast-growing interests in building AI systems with such human-like reasoning capabilities based on inferential commonsense knowledge (Storks et al., 2019; Bosselut and Choi, 2019). Such systems are often evaluated by answering questions based on narratives. As illustrated in Figure 1, given a narrative "Austin often spends her weekend at the lake fishing with friends" and a question regarding the intention of Austin, an AI system is supposed to associate this event to relevant events in an inferential knowledge base or a web-scale corpus, find plausible reasons of those events, and conclude that "wanted to relax" is the most probable answer.

Knowledge-based approaches to such commonsense question-answering (QA) require an inferential knowledge base. ATOMIC (Sap et al., 2019a) is the largest commonsense knowledge base of this kind, which contains 300,000 short textual description of events and 877,000 typed *if-then* relations between events, categorised into 9 dimensions. For instance, *IF* the event "X puts trust in Y" occurs and the target relation is "xWant", *THEN* "X wants to develop a relationship". Prior work utilizes knowledge in ATOMIC by formulating the learning problem as event prediction in *if-then* relations (Bosselut et al., 2019). In particular, they encode the textual description of an event and a relation into an embedding, and maximise the probability of predicting the description of the associated event or characteristic of an involved person. However, due to the nature of commonsense knowledge, given an event and a relation, there are multiple plausible associated events. Moreover, these models fail to predict all associated events

Proceedings of the 28th International Conference on Computational Linguistics, pages 5347–5359
Barcelona, Spain (Online), December 8-13, 2020

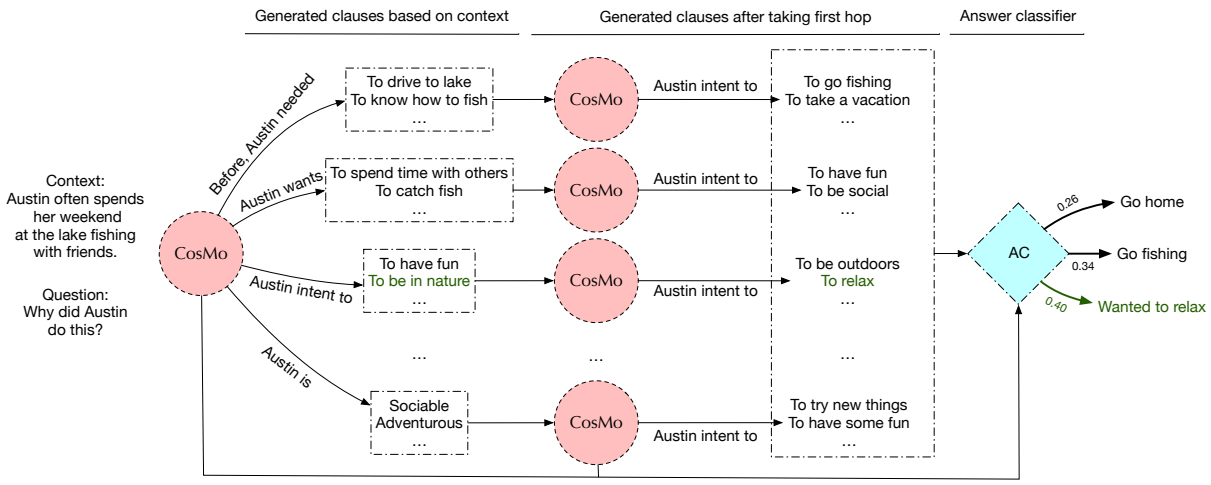

Figure 1: The illustration of the process of answering a question with our proposed model. The model receives a context and a question. Using Conditional SEQ2SEQ-based Mixture Model (COSMO), clauses based on the context and different relations are generated. The generated clauses are then used to generate new set of clauses based on the question. The Answer Classifier (AC) module selects the answer with higher score based on generated clauses at the last step and scores of COSMO. The path to the correct answer is distinguished with green color.

to a given event and relation, hence fail to identify every implicit reasoning path to address the task of question answering.

To address the above challenges, we propose a conditional SEQ2SEQ-based mixture model, named COSMO, to answer questions on everyday inferential knowledge in a *zero-shot* setting. As the events in our target commonsense KGs are described in text, such as ATOMIC, we distil the knowledge into a Sequence-to-Sequence (SEQ2SEQ) model. The distilled model memorises knowledge of the KG and generalise it to new events for a given context and specific relation, on-demand. However, a direct application of maximum-likelihood training on SEQ2SEQ models leads to deteriorated performance, because the underlying distribution over *diverse* outputs is inherently *multi-modal* (Shen et al., 2019). To address the above challenge, we incorporate a latent variable into a pre-trained SEQ2SEQ transformers (Yan et al., 2020). Each value of the variable corresponds to an embedding, which indicates the hidden factors explaining different hypothesis. Unlike the existing training methods of such models that cannot guarantee a definitive alignment of different hypothesis with different components during training, our proposed constrained-EM (expectation-maximisation) training procedure enforces that for the same input, different model outputs align with different latent embedding.

To tackle the task of Commonsense QA, we use COSMO to generate context-dependent information, for desired relations. The new generated events form a dynamic knowledge graph, which is reasoned over, until the correct answer is chosen by the model. The ubiquitous nature of everyday commonsense knowledge and lack of training data for each situation motivate addressing the task in zero-shot setting. It is unrealistic to expect the presence of manually constructed training datasets in each commonsense QA domain. With the lack of training data in such question-answering settings, we devise a bespoken answer scoring module to assess the likelihood of each answer. Given a narrative and a question, the model generates sequences of plausible fact descriptions, as *reasoning paths*, by choosing different values of the latent variable. For each answer and each reasoning path, the scoring module computes a score based on similarity between the answer and the last fact description in the reasoning path. The most probable answer is determined by its answer score and the probability of the associated reasoning path *jointly*.

To sum up, our contributions are three-folds:

- We propose COSMO, a conditional SEQ2SEQ-based mixture model for zero-shot question answering on inferential knowledge. COSMO constructs a dynamic KG on-demand, which is used to reason over and answer commonsense questions.

- We propose a novel training procedure to enforce hard alignments between latent variables and diverse hypotheses, to ensure the diversity of the associated KG events distilled in the SEQ2SEQ.
- Our experimental results on SocialIQA (Sap et al., 2019b) show that our model achieves superior accuracy and significantly more diverse hypotheses than competitive baselines.

2 Related Works

Commonsense Knowledge Bases Introducing Commonsense Knowledge Graphs (KG) has provided a source of information for machines on the task of commonsense reasoning. This KGs, encode commonsense relations between different pair of events/concepts. Speer et al. (2017) assembled a KG from a variety of sources, ConceptNet, which represents general knowledge in form of tuples. While ConeptNet is more centered around taxonomic knowledge, Sap et al. (2019a) constructed ATOMIC, a KG consisting inferential knowledge. Information in ATOMIC (Sap et al., 2019a) are presented as *if-then* relationship between events, mental states, and persona. The information in ATOMIC includes such information for the agent of the event and others, who might be affected. Zhang et al. (2020) proposed an automatic approach for collecting eventuality KG, ASER, consisting events, activities, and states. Although the presented KGs provide rich commonsense information, reasoning about social situation requires a dynamic on-demand approach for context-dependent information generation.

Commonsense Knowledge Base Completion As an essential way of enabling machines to perform commonsense reasoning, the methods for automatic KG construction and completion have been studied recently. Sap et al. (2019a) used LSTM as a generator for commonsense knowledge about social situations. Davison et al. (2015) proposed an unsupervised method to extract commonsense knowledge from pre-trained models, to complete KG. Malaviya et al. (2020) developed a model which takes both structural and semantic characteristics of the nodes in a KG to address this task. On the other hand, some works have developed generative models on top of pre-trained language models to extract new commonsense information (Bosselut et al., 2019; Malaviya et al., 2019). However, when adapting to the KG, the previously acquired knowledge of these models is forgotten. Unlike these approaches, our neural model ensures both diversity and memorisation.

Commonsense Question Answering Recent surge of commonsense question answering dataset (Sap et al., 2019b; Zellers et al., 2018) has led to many supervised approaches to address this task. Most of these approaches are based on transfer learning, where a large scale pre-trained model (Lan et al., 2019; Devlin et al., 2018) is finetuned on the target task(Sap et al., 2019b). On the other hand, with the release of new commonsense KGs, such as ATOMIC (Sap et al., 2019a) and ConceptNet (Speer et al., 2017), the possibility of enriching language models with these KGs has been investigated vastly (Lv et al., 2020; Mitra et al., 2019; Banerjee and Baral, 2020). Most of these approaches map the context of a question to an entity/event in KG and perform reasoning on the KG (Weissenborn et al., 2017; Paul and Frank, 2019; Lin et al., 2019). While these methods enable the system to perform multi-hop reasoning, they are limited to the set of entities in KG. Other works deployed generative models to generate context-aware information to answer the questions (Shwartz et al., 2020; Bosselut and Choi, 2019; Banerjee and Baral, 2020). These approaches overcome the limitation of static KGs, but they lack diversity in generating new relations, which makes their inference path limited. Furthermore, for scoring the answer choices they have solely relied on the conditional likelihood of the generative model, which lack to perform when the distribution of KG and QA differ.

Zero-shot Question Answering In recent years, zero-shot learning has become a popular method to conquer the inability of machine learning system to perform on unobserved data (Wang et al., 2019). While this method has been thoroughly researched in other fields (Zhao et al., 2019), the necessity of using such approach has inspired many works in question answering systems as well. Visual Question Answering (Teney and Hengel, 2016) adopted a zero-shot learning approach to extract features from unseen text description about given images. Lewis (2019) proposed an unsupervised extractive question answering model by using unsupervised data generation for converting the QA task to a cloze translation task.

Puri (2020) improved the quality of unsupervised extractive question answering systems by introducing an approach involving answer generation, question generation and roundtrip filtration. In addition, Li et al. (2020) used Wikipedia's data in order to overcome the drawback of Lewis et al. (2019). Compared to the previous question answering model, our model shows the novelty in generation new clauses from given contexts.

3 Our Approach

In this section, we present our model COSMO for question answering in the zero-shot setting. In this setting, there is no training data for the QA task, thus we train our model only on ATOMIC to acquire inferential knowledge. We apply the trained model to answer multi-choice questions on everyday inferential knowledge by augmenting the trained model with a non-parametric answer scoring module.

Formally, given a narrative describing a context $\boldsymbol{c}$, a commonsense-required question $\boldsymbol{q}$, and a set of m candidate answers $\mathcal{A} = \{\boldsymbol{a}^0, ..., \boldsymbol{a}^m\}$, the task is to choose the most plausible answer from the set $\mathcal{A}$. In order to measure the plausibility of answer candidates, the required commonsense inferential knowledge is from a large external knowledge base $\mathcal{B}$, which is a set of typed *if-then* relations. Each *if-then* relation takes the form of "*if* $\boldsymbol{z}_i$ *and* r *then* $\boldsymbol{z}_j$", where $\boldsymbol{z}_i$ denotes a word sequence describing that the event i and r is a relation between events from the pre-defined set $\mathcal{R}$. Such a relation is also referred to as inference dimension in ATOMIC.

To this end, we define the target task as finding the most plausible answer by applying inferential reasoning over the knowledge base $\mathcal{B}$. The reasoning process is characterised by finding a plausible reasoning path $\boldsymbol{Z}$, which starts from a given context $\boldsymbol{c}$, along the target relation determined by a question $\boldsymbol{q}$, to reach an answer $\boldsymbol{a}$. A reasoning path is a sequence of events correlated by *if-then* relations derived from $\mathcal{B}$. Therefore, for a given context $\boldsymbol{c}$ and a question $\boldsymbol{q}$, we find the most likely answer by solving the following optimisation problem:

$$\arg \max_{\boldsymbol{a} \in \mathcal{A}, \boldsymbol{Z} \in \mathcal{Z}} \log Pr(\boldsymbol{a}|\boldsymbol{Z}) Pr(\boldsymbol{Z}|\boldsymbol{c}, r, \mathcal{B}) Pr(r|\boldsymbol{q}) \tag{1}$$

where $Pr(r|\boldsymbol{q})$ is the probability of a target relation r given the question [1]; $Pr(\boldsymbol{a}|\boldsymbol{Z})$ denotes an *answer scoring module* estimating the probability of an answer $\boldsymbol{a}$ given a reasoning path; $Pr(\boldsymbol{Z}|\boldsymbol{c}, r, \mathcal{B})$ is the probability of a reasoning path $\boldsymbol{Z}$, conditioned on the context $\boldsymbol{c}$, the target relation r, and the knowledge base $\mathcal{B}$. The local distribution $Pr(\boldsymbol{Z}|\boldsymbol{c}, r, \mathcal{B})$ can be further factorized into

$$Pr(\boldsymbol{Z}|\boldsymbol{c}, r, \mathcal{B}) = Pr(\boldsymbol{z}_0|\boldsymbol{c}, r, \mathcal{B}) \prod_{t=1}^{T} Pr(\boldsymbol{z}_t|\boldsymbol{z}_{<t}, \boldsymbol{c}, r, \mathcal{B}) \tag{2}$$

$$= Pr(\boldsymbol{z}_0|\boldsymbol{c}, r, \mathcal{B}) \prod_{t=1}^{T} Pr(\boldsymbol{z}_t|\boldsymbol{z}_{t-1}, r, \mathcal{B}) \tag{3}$$

where $Pr(\boldsymbol{z}_0|\boldsymbol{c}, r, \mathcal{B})$ is the distribution of the first event given a context and a target relation and $Pr(\boldsymbol{z}_t|\boldsymbol{z}_{<t}, \boldsymbol{c}, r, \mathcal{B})$ characterises the distribution of future events. To simplify inferential reasoning, we assume $\boldsymbol{z}_t$ with $t > 0$ is conditionally independent of $\boldsymbol{c}$ and $\boldsymbol{z}_{<t-1}$ such that both $Pr(\boldsymbol{z}_t|\boldsymbol{z}_{t-1}, r, B)$ and $Pr(\boldsymbol{z}_0|\boldsymbol{c}, r, \mathcal{B})$ can be estimated by the same module, which predicts a future event by taking a textual description and the target relation as input. The module is referred to as the ***KB module*** because it is trained on ATOMIC to acquire inferential knowledge.

In the following, we will detail the KB module as well as its training procedure on ATOMIC, followed by presenting the answer scoring module and how to apply them together for the target task.

3.1 KB Module for Inferential Reasoning

The KB module aims to encode *if-then* relations in the KB $\mathcal{B}$ into model parameters, and apply the knowledge to infer future events given a target relation and a text describing a current event or a context.

[1] Due to the regularity of questions in the benchmark dataset, we can directly apply rules to find out the target relations. Thus $Pr(r|\boldsymbol{q})$ is always one in our experiments.

As both inputs and outputs are word sequences, we formulate the task of event prediction as a sequence to sequence prediction problem. As a result, we are able to exploit the powerful pre-trained transformer based SEQ2SEQ models (Devlin et al., 2018; Yan et al., 2020) as the backbone.

The key challenge of using pre-trained SEQ2SEQ models on ATOMIC is the diversity of output events based on a given event z_i and relation r. The key idea herein is to align different latent factors h_k with different outputs for the same input. Each latent factor is the value of a latent variable for alignment. After introducing the latent variable, we obtain a conditional mixture model of the following form:

$$Pr(\boldsymbol{z}|\boldsymbol{x}, r; \boldsymbol{\theta}_{\mathcal{B}}) = \sum_{k=1}^{K} Pr(\boldsymbol{z}, h_k|\boldsymbol{x}, r; \boldsymbol{\theta}_{\mathcal{B}}) = \sum_{k=1}^{K} Pr(\boldsymbol{z}|h_k, \boldsymbol{x}, r; \boldsymbol{\theta}_{\mathcal{B}}) Pr(h_k|\boldsymbol{x}) \tag{4}$$

where $\boldsymbol{x}$ denotes either the description of an event or a context. Here we replace $\mathcal{B}$ with the model parameters $\boldsymbol{\theta}_{\mathcal{B}}$ as the knowledge-base is encoded into the model parameters. We further assume $Pr(h_k|\boldsymbol{x})$ follows a uniform distribution during prediction because target QA datasets follow a more different distribution than ATOMIC.

As each latent variable value can be represented by a symbol, the module for $Pr(\boldsymbol{z}|\boldsymbol{h}_k, \boldsymbol{x}, r)$ is realized by a SEQ2SEQ model. It takes as input a token sequence consisting of a latent variable value $\boldsymbol{h}_k$, a word sequence $\boldsymbol{x}$, and a relation symbol r, and predicts a word sequence representing the next event $\boldsymbol{z}$. We enrich the input vocabulary of the chosen SEQ2SEQ model with the symbols of h_k and r which are mapped to the corresponding latent embeddings and relation embeddings during forward propagation.

We select ProphetNet (Yan et al., 2020) as the SEQ2SEQ backbone model, because it achieves the best performance on a number of natural language generation tasks. The encoder and the decoder of this model utilize n-stream self-attention mechanism and future n-gram prediction in order to encourage planning for the future tokens and prevent overfitting on strong local correlations.

Training The goal of training is to learn the parameters of the following model on ATOMIC,

$$\max_{\boldsymbol{\theta}_{\mathcal{B}}} \prod_{r(\boldsymbol{x},\boldsymbol{z})\in\mathcal{B}} \sum_{k=1}^{K} Pr(\boldsymbol{z}|h_k, \boldsymbol{x}, r; \boldsymbol{\theta}_{\mathcal{B}}) Pr(h_k|\boldsymbol{x}). \tag{5}$$

More specifically, we train the model on each *if-then* relation of the form "*if* $\boldsymbol{x}$ *and* r *then* $\boldsymbol{z}$" in ATOMIC by taking $\boldsymbol{x}$ and r as input and predicting $\boldsymbol{z}$.

Prior work on diverse machine translation (He et al., 2018a; Shen et al., 2019; Cho et al., 2019) suggests to apply online hard EM by interleaving the following two steps for each mini-batch.

- **E-step**: estimate the value of the latent variable through $\hat{k} = \arg\max_k Pr(\boldsymbol{z}|h_k, \boldsymbol{x}, r; \boldsymbol{\theta}_{\mathcal{B}})$ using the current parameters $\boldsymbol{\theta}_{\mathcal{B}}$.
- **M-step**: The model parameters $\boldsymbol{\theta}_{\mathcal{B}}$ are then updated by minimising the cross-entropy loss on $Pr(\boldsymbol{z}|h_{\hat{k}}, \boldsymbol{x}; \boldsymbol{\theta}_{\mathcal{B}})$.

However, the greedy search in the E-step may still assign the same latent variable value to different target sequences. To eliminate the problem, we modify and constrain the E-Step by requiring that, different target sequences of the same input need to be assigned different latent variable values. More specifically, for each output set $S_{r,\boldsymbol{x}} := \{j \mid \exists\, r(\boldsymbol{x}, \boldsymbol{z}_j) \in \mathcal{B}\}$ sharing the same input $\boldsymbol{x}$ and r in a mini-batch, we tackle this problem by solving the following combinatorial optimization problem.

$$\begin{aligned} &\max_{\{u_{j,k}\}} \sum_j \sum_k u_{j,k} \log Pr(\boldsymbol{z}_j|h_k, \boldsymbol{x}, r; \boldsymbol{\theta}_{\mathcal{B}}) \\ &s.t. \sum_k \sum_j u_{j,k} = |S_{r,\boldsymbol{x}}| \\ &\sum_{j\in S_{r,\boldsymbol{x}}} u_{j,k} \le 1,\ \forall k \end{aligned}$$

where $u_{j,k}$ is a binary variable indicating the alignment between $\boldsymbol{z}_j$ and h_k, and $|S_{r,\boldsymbol{x}}|$ is the number of target events in the output set. Here we set K always larger than $|S_{r,\boldsymbol{x}}|$.

We solve the above problem by a heuristic-based search. We compute the log probability of $Pr(\boldsymbol{z}_j|h_k, \boldsymbol{x}, r; \boldsymbol{\theta}_{\mathcal{B}})$ for all combination of k and j. Then we sort those log probabilities, and select $u_{j,k}$ satisfying the hard constraints in order. More details about the algorithm can be found in the pseudo code in Algorithm 1.

Algorithm 1: Conditional Mixture Model

input : source $\boldsymbol{x}$, target $\{\boldsymbol{z}_j\}_{j=1}^{J}$, relation r, latent variable $\{h_k\}_{k=1}^{K}$ $(K > J)$
output : Model parameter $\theta_{\mathcal{B}}$

empty list Γ
while $j < M$ **do**
  while $k < K$ **do**
    $l_{j,k} := p(\boldsymbol{z}_j|h_k, \boldsymbol{x}, r, ; \theta_{\mathcal{B}})$;
    Γ := add ($\boldsymbol{x}$,r,$\boldsymbol{z}_j$, h_k,$l_{j,k}$) to Γ;
  end while
end while
Γ := Sort(Γ) based on values of $l_{j,k}$
create empty lists Δ, ζ, ι
repeat
  for $\boldsymbol{x}$,r,$\boldsymbol{z}_j$, h_k,$l_{j,k} \in \Gamma$ **do**
    if $\boldsymbol{z}_j \notin \zeta$ ***and*** $h_k \notin \iota$ **then** add $(\boldsymbol{x}, r, \boldsymbol{z_j}, l_k)$ to Δ, add $\boldsymbol{z}_j$ to ζ, add h_k to ι ;
  end for
until *every* $\boldsymbol{z}_j$ *is assigned with a* h_k;
for $\boldsymbol{x}, r, \boldsymbol{z_j}, l_k \in \Delta$ **do**
  Make forward and backward propagation w.r.t the training objective (Equation 5)
  Update the model parameter (θ)
end for
return θ

3.2 Answer Scoring Module

The main objective of the answer scoring module in Eq. (1) is to assign a score to each answer candidate. For this purpose, Bosselut (2019) proposed an averaged word probability approach. In this method, event prediction is considered as language modelling task, hence the score is defined as the average probability of generating each token of an event. However, this method is not theoretically grounded and largely relies on heuristics. Based on the probabilistic model in Eq. (1), the true answers should be semantically similar to the last events in plausible reasoning paths derived from contexts and questions. Thus, the answer scoring module solely depend on the last events of reasoning paths.

$$Pr(\boldsymbol{a}|\boldsymbol{Z}) = Pr(\boldsymbol{a}|\boldsymbol{z}_T) \tag{6}$$

where $\boldsymbol{z}_T$ denotes the event generated at time step T. As the distribution is characterised by semantic similarity between the last events and answer candidates, we define the distribution as:

$$Pr(\boldsymbol{a}|\boldsymbol{z}_T) = \frac{\exp(-\gamma d(\boldsymbol{a}, \boldsymbol{z}_T))}{\sum_{\boldsymbol{a'} \in \mathcal{A}} \exp(-\gamma d(\boldsymbol{a'}, \boldsymbol{z}_T))} \tag{7}$$

where $d(\boldsymbol{a}, \boldsymbol{z})$ is a distance function between an answer and an event, and γ is a hyperparameter adjusting the temperature.

After distilling the SEQ2SEQ model with information of ATOMIC, we plug the trained KB module and the answer module into the model. COSMO answers questions by applying Eq. (1). We apply beam search to find top-k reasoning paths for each latent variable value up to a pre-specified number of hops T. The most plausible answer is the most probable reasoning path, whose last event achieves the highest similarity with the answer.

4 Experiments

In this section, we report the evaluation of our model on zero-shot question answering. To this end, we use test set and development set of SocialIQA (Sap et al., 2019b). We evaluate the performance of our model with two variations. In zero-hop setup, we only consider generating clauses using COSMO based on the context of the question, and the answers are then scored against the generated clauses. In one-hop setting, we take a step further and generate more clauses using the generated clauses in the previous step. This approach helps us uncover more implicit context-dependent information. The final answer is chosen against the combination of all generated clauses. To further analyse the capability of our proposed model in terms of clause generation, we compare our model to other approaches on ATOMIC (Sap et al., 2019a), and test and development set of SocialIQA.

4.1 Datasets

SocialIQA This dataset consists of commonsense questions, which aims to evaluate a model's capability in inferring implicit social context. Each question in this dataset is presented with a context, which describes the situation, and three answer candidates. For the purpose of addressing this task, we convert each question to one of the relations in the KG, using a pattern-based system. The details of this module is provided in the Appendix 6.1. This dataset contains a total of 37,588 questions. However, in a zero-shot setting we only use the development and test set for evaluation, where they contain 1,954 and 2,224 questions, respectively.

ATOMIC This dataset consists of 877K sets of subject, relation, and object, where each set describes a social commonsense situation. The subjects are an event (e.g., "PersonX puts trust in PersonY"), which poses a social situation. The relations are categorised into 9 dimensions (e.g., xEffect). The object is indicated by the relation, which shows the causes of subject, the effects of subject on the agent, and others, or attributes of the agent. The original data split, 710K/80K/87K for train/development/test, by Sap (2019a) is used in our experiments.

4.2 Baselines

For evaluating our proposed zero-shot question answering model, we compare COSMO to the state-of-the-art pretrained language models, GPT (Radford et al., 2018). Also, we consider different variation of GPT-2 (Radford et al., 2019), including GPT2-117M, GPT2-345M, and GPT2-762M. For this purpose the questions are converted to state sentence (as described in Appendix B). The language model scores the answers based on cross-entropy loss of concatenation of context, question, and answers. In addition, we compare our model to two variation of COMET-CA and COMET-CGA (Bosselut and Choi, 2019). We also report the performance of the state-of-the-art supervised methods, Bert (Devlin et al., 2018) and RoBERTa (Liu et al., 2019).

To further analyse the performance of our model in generation of clauses, we compare our model to the state-of-the-art automatic knowledge base completion model, that has been used in zero-shot commonsense question answering task, COMET (Bosselut et al., 2019). Also, to assess the diversity, we consider comparison of our proposed model to the SEQ2SEQ model presented in section 3, without applying latent variable (ProphetNet)(Yan et al., 2020), and state-of-the-art model for applying the latent variable (MoE)(Shen et al., 2019).

4.3 Metrics

For evaluating the performance of models in zero-shot question answering task, we report the accuracy of each model in choosing the correct answer. In addition, to compare the effectiveness of our proposed scoring function, we compare three variation of our scoring function to two different scoring functions proposed in (Bosselut and Choi, 2019). Furthermore, having a diverse clause generation is an advantage of our proposed model. As a quantitative evaluation, for a set of clause generated by the model denoted as $\{\hat{y}\}_{m=1}^{M}$, we use div_bleu and div_ngram (He et al., 2018b), which are defined as follow:

$$\text{div\_bleu} = 1 - (\sum_{i=1}^{M} \sum_{j=i+1}^{M} \text{BLEU}(\hat{y}_i, \hat{y}_j))/M(M-1)/2 \quad (8)$$

$$\text{div\_ngram} = 1 - \frac{|\cap_{m=1}^{M} \text{ngram}(\hat{y_m})|}{|\cup_{m=1}^{M} \text{ngram}(\hat{y_m})|} \quad (9)$$

where ngram(y) indicates the set of unique ngrams in a sequence y. For each model the top-50 clause generated by beam search is considered for evaluation purposes. For div_bleu we report the average result of BLEU-1, BLEU-2, BLEU-3, and BLEU-4, with Smoothing1 (Chen and Cherry, 2014). For div_ngram, we report the average results of 1-gram, 2-gram, 3-gram, and 4-gram.

4.4 Experimental Details

For training our proposed Knowledge Graph Neuralisation model, COSMO [2], we finetune the SEQ2SEQ model using the pretrained model of Yan (2020). Our implementation is based on FAIRSEQ [3]. The model consists of 12 layers of encoder and 12 layers of decoder. The embedding size and batch size are set to 1024 and 512, respectively. The number of future ngram is set to 2. We use Adam optimiser (Kingma and Ba, 2014) with a peak learning rate of 1×10^{-4}. For the question answering module, we consider answering without taking a hop, and taking one-hop. Since we evaluate our model in zero-shot setting, the γ in equation 7 is set to one, and we use cosine similarity function as the distance function. At each step, we consider beam-10 for clause generation.

4.5 Results and Discussions

Zero-shot Commonsense Question Answering Table 1 summarises the result of performance of different models in task of commonsense question answering. The performance of the models in zero-shot setting is reported in the top section of the table. It's clear that our proposed model outperforms the other methods, in both development and test set, by up to +5.2%. Furthermore, the improvement over taking a hop in answering question, suggests that the implicit information related to a given context can help answering some questions. However, the existing gap with supervised models, indicates potential possibility of improvement.

	MODEL	Dev Acc.	Test Acc.
Unsupervised	Random	33.3	33.3
	GPT (Radford et al., 2018)	41.8	41.7
	GPT2-117M (Radford et al., 2019)	40.7	41.5
	GPT2-345M (Radford et al., 2019)	41.5	42.5
	GPT2-762M (Radford et al., 2019)	42.5	42.4
	COMET-CA (Bosselut and Choi, 2019)	48.7	49.0
	COMET-CGA (Bosselut and Choi, 2019)	49.6	51.9
	COSMO (zero-hop)	52.9	53.1
	COSMO (one-hop)	**54.8**	**55.0**
Supervised	BERT-Large (supervised) (Devlin et al., 2018)	66.0	66.4
	RoBERTa (supervised) (Liu et al., 2019)	76.6	77.8
	human	86.9	84.4

Table 1: The accuracy of answer prediction of our proposed model compared to the state-of-the-art models on SocialIQA, on development and test set.

[2]Code available at https://github.com/farhadmfar/cosmo

[3]https://github.com/pytorch/fairseq

Table 2 summarises the results of evaluation of different methods of scoring the answer candidates. The scoring function proposed by Bosselut (2019), averaged word probability, comes in two variations. To overcome the answer imbalance given specific questions, they propose adding Pointwise Mutual Information of question and answers to the original scoring function. The distance function (Eq. (1)) in our proposed model is evaluated with three variations of directly using probability of SEQ2SEQ model (SEQ2SEQ), BLEU function, and cosine distance. The results indicate that using cosine distance function improves the results by +4.93% and +6.48% over the second best approach, on development and test set, respectively. The experiments suggest that taking the similarity of clause generations at each step with the answers provides stronger classifier over answers, compared to considering only the probability of the generative models. The lack of performance of the latter configuration can be rooted in training phase of generative models, where some phrases, regardless of the context, are seen frequently together, resulting in achieving higher probability by the model.

MODEL	Dev Acc.	Test Acc.
averaged word probability (Bosselut and Choi, 2019)	36.59	33.67
averaged word probability (without pmi) (Bosselut and Choi, 2019)	34.98	35.05
COSMO (SEQ2SEQ)	34.98	35.05
COSMO (BLEU)	47.93	46.62
COSMO (Cosine)	**52.86**	**53.1**

Table 2: The results of using our proposed answer classifier function compared to the baselines, on development and test set of SocialIQA.

Diverse Clause Generation One of the strength of our proposed model is the ability to generate diverse clauses given a subject and a relation. Table 3 shows the result of diverse generation in terms of div_ngram and div_bleu. As it can be seen, for div_ngram, our proposed model achieves the highest performance on test set of ATOMIC. Furthermore, we observe that our model outperforms the baseline methods on development and test set of SocialIQA.

	MODEL	ATOMIC	SoicalIQA Dev.	SocialIQA Test
div_ngram	COMET (Bosselut et al., 2019)	43.19	41.50	39.91
	ProphetNet (Yan et al., 2020)	67.27	49.36	49.39
	MoE (Shen et al., 2019)	75.61	71.79	71.19
	COSMO	**80.72**	**79.21**	**79.09**
div_bleu	COMET (Bosselut et al., 2019)	84.22	84.03	79.44
	ProphetNet (Yan et al., 2020)	82.44	77.58	74.55
	MoE (Shen et al., 2019)	89.60	88.33	87.90
	COSMO	**93.03**	**92.34**	**92.39**

Table 3: The results of div_ngram (top section) and div_bleu(bottom section) of our model compared to the baselines, on test set of ATOMIC and SocialIQA.

The results of div_bleu also shows that on ATOMIC and SocialIQA development and test set, our model outperforms all the baselines on all variation of BLEU function. The results suggest the capability of our proposed model in diverse clause generations.

4.6 Qualitative Analysis

In this section, we demonstrate the capability of our model on diverse clause generation, and its effect on commonsense question answering. Table 4 provides two example from test set of SocialIQA. For each example, top-5 clauses generated by our proposed model and COMET, given context and question, are provided. Both examples show capability of our model in generating divers outputs, which results in finding the correct answer.

Example 1	*Context*: Alex is a store owner and observed every person's contribution carefully. Alex rewarded every person accordingly. *Question*: Why did Alex do this? *Answers*: contribute to the local community, close his store soon, reward more for more deserving persons *Correct Answer*: reward more for more deserving persons
COMET	to be a good employee to be helpful to be a good salesperson to make sure everything goes smoothly to be a good citizen
COSMO	to be fair to reward good work to appreciate good work to reward good people to show appreciation
Example 2	*Context*: Bailey was a nice person so she called the family together. *Question*: What will happen to Others? *Answers*: talk to the family, hate bailey, thank bailey *Correct Answer*: thank bailey
COMET	family members talk to Bailey family members get to know Bailey the family members get to know Bailey better the family members talk to Bailey about Bailey spend time with Bailey
COSMO	the family members respect Bailey enjoy Bailey's company the family thank Bailey the family respects Bailey the people interact with Bailey

Table 4: Examples of the test set of SocialIQA, with the clause generated by our proposed model, COSMO, compared to COMET.

5 Conclusion

In this paper, we propose a novel approach for neuralising large-scale commonsense knowledge graph, Conditional SEQ2SEQ-based Mixture Model, COSMO. Our proposed model provides diverse clause generation, to ensure coverage for the target task. We use the proposed model to generate diverse context related clauses, alongside with our proposed answer classifier model, to address the task of zero-shot commonsense question answering task. Empirical results on zero-shot commonsense question answering dataset show the superiority of our model over the state-of-the-art methods, by up to 5.2%. Furthermore, our model outperforms baselines in terms of diversity of clause generations.

References

Pratyay Banerjee and Chitta Baral. 2020. Knowledge fusion and semantic knowledge ranking for open domain question answering. *arXiv preprint arXiv:2004.03101*.

Antoine Bosselut and Yejin Choi. 2019. Dynamic knowledge graph construction for zero-shot commonsense question answering. *arXiv preprint arXiv:1911.03876*.

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

Boxing Chen and Colin Cherry. 2014. A systematic comparison of smoothing techniques for sentence-level bleu. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367.

Jaemin Cho, Min Joon Seo, and Hannaneh Hajishirzi. 2019. Mixture content selection for diverse sequence generation. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3119–3129. Association for Computational Linguistics.

Ernest Davis and Gary Marcus. 2015. Commonsense reasoning and commonsense knowledge in artificial intelligence. *Communications of the ACM*, 58(9):92–103.

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*.

Xuanli He, Gholamreza Haffari, and Mohammad Norouzi. 2018a. Sequence to sequence mixture model for diverse machine translation. In Anna Korhonen and Ivan Titov, editors, *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, pages 583–592. Association for Computational Linguistics.

Xuanli He, Gholamreza Haffari, and Mohammad Norouzi. 2018b. Sequence to sequence mixture model for diverse machine translation. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 583–592.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. 2019. Unsupervised question answering by cloze translation. *arXiv preprint arXiv:1906.04980*.

Zhongli Li, Wenhui Wang, Li Dong, Furu Wei, and Ke Xu. 2020. Harvesting and refining question-answer pairs for unsupervised qa. *arXiv preprint arXiv:2005.02925*.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2822–2832.

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

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering.

Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. 2019. Exploiting structural and semantic context for commonsense knowledge base completion. *arXiv preprint arXiv:1910.02915*.

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

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*.

Debjit Paul and Anette Frank. 2019. Ranking and selecting multi-hop knowledge paths to better predict human needs. In *Proceedings of NAACL-HLT*, pages 3671–3681.

Raul Puri, Ryan Spring, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2020. Training question answering models from synthetic data. *arXiv preprint arXiv:2002.09599*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

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 *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3027–3035.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019b. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*.

Tianxiao Shen, Myle Ott, Michael Auli, and Marc'Aurelio Ranzato. 2019. Mixture models for diverse machine translation: Tricks of the trade. In *International Conference on Machine Learning*, pages 5719–5728.

Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Unsupervised commonsense question answering with self-talk. *arXiv preprint arXiv:2004.05483*.

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

Shane Storks, Qiaozi Gao, and Joyce Y Chai. 2019. Commonsense reasoning for natural language understanding: A survey of benchmarks, resources, and approaches. *arXiv preprint arXiv:1904.01172*, pages 1–60.

Damien Teney and Anton van den Hengel. 2016. Zero-shot visual question answering. *arXiv preprint arXiv:1611.05546*.

Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. 2019. A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–37.

Dirk Weissenborn, Tomáš Kočiskỳ, and Chris Dyer. 2017. Dynamic integration of background knowledge in neural nlu systems. *arXiv preprint arXiv:1706.02596*.

Yu Yan, Weizhen Qi, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. *arXiv preprint arXiv:2001.04063*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *arXiv preprint arXiv:1808.05326*.

Hongming Zhang, Xin Liu, Haojie Pan, Yangqiu Song, and Cane Wing-Ki Leung. 2020. Aser: A large-scale eventuality knowledge graph. In *Proceedings of The Web Conference 2020*, pages 201–211.

Wei Zhao, Haiyun Peng, Steffen Eger, Erik Cambria, and Min Yang. 2019. Towards scalable and reliable capsule networks for challenging nlp applications. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1549–1559.

6 Appendix

6.1 Appendix A

	Question	Associated Relation	Assoicated phrase
cause	Why did AGENT do this?	xIntent	AGENT did this because ...
	What does AGENT need to do before this?	xNeed	Before, AGENT needs to ...
attribute	How would you describe AGENT?	xAttr	AGENT is ...
effect	How would AGENT feel afterwards?	xReact	AGENT feels ...
	What will AGENT want to do next?	xWant	After, AGENT wants to ...
	What will happen to AGENT?	xEffect	The effect on AGENT will be ...
	How would others feel as a results?	oReact	Others feel ...
	What will others do next?	oWant	After, others will want to ...
	What will happen to others?	oEffect	The effect on other will be ...

Table 5: The templates that have been used to map the question from SocialIQA to the relations of ATOMIC, and phrases to use for language models, categorised by relation type (cause/effect/attribute). "AGENT" refers to the person who is the subject of the given context in the question answering tuple.