DimonGen: Diversified Generative Commonsense Reasoning for Explaining Concept Relationships

Chenzhengyi Liu* **Jie Huang*†** **Kerui Zhu** **Kevin Chen-Chuan Chang**
University of Illinois at Urbana-Champaign, USA
{cl115, jeffhj, keruiz2, kcchang}@illinois.edu

Abstract

In this paper, we propose *DimonGen*, which aims to generate diverse sentences describing concept relationships in various everyday scenarios. To support this, we first create a benchmark dataset for this task by adapting the existing CommonGen dataset. We then propose a two-stage model called MoREE to generate the target sentences. MoREE consists of a mixture of retrievers model that retrieves diverse context sentences related to the given concepts, and a mixture of generators model that generates diverse sentences based on the retrieved contexts. We conduct experiments on the DimonGen task and show that MoREE outperforms strong baselines in terms of both the quality and diversity of the generated sentences. Our results demonstrate that MoREE is able to generate diverse sentences that reflect different relationships between concepts, leading to a comprehensive understanding of concept relationships.[1]

1 Introduction

Concepts are mental representations of classes or categories of objects, events, or ideas, distinguished by shared characteristics that set them apart from other things. For instance, the concept of "dog" represents a class of animals that share characteristics such as being four-legged, having fur, and being domesticated. These concepts are crucial in helping us understand and communicate about the world around us.

To fully grasp concepts, it is important to understand the relationships between them. Researchers have proposed using generated sentences as a means to model these relationships more effectively (Lin et al., 2020; Huang et al., 2022a,c; Huang and Chang, 2022b). For example, CommonGen (Lin et al., 2020) aims to generate coherent sentences that describe everyday scenarios involving specific sets of common concepts, while Open Relation Modeling (Huang et al., 2022a) generates informative sentences that describe relationships between concepts/entities.

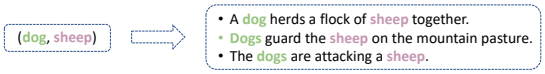

Figure 1: An example of DimonGen. The input is a pair of concepts and the output is a set of sentences that capture different ways in which these concepts interact.

However, in real-world scenarios, concepts often refer to broad classes, and their relationships can be complex. This can make it challenging to summarize these relationships through a single sentence. For example, "dog" and "sheep" are both animal concepts, but while "dogs" can herd "sheep", they can also attack them. A single sentence would not accurately convey this complexity, leading to an insufficient understanding. Additionally, this approach can also introduce bias, particularly when concepts are related to sensitive topics such as gender or race. For instance, the statement "*women* are better suited for caregiving roles than *men*." is a biased statement.

To mitigate the above issues, we propose a new task called *DimonGen: Diversified Generative Commonsense Reasoning*. The task involves generating diverse sentences that describe the relationships between two given concepts, such as the example shown in Fig. 1 of the concept pair "dog" and "sheep". This helps build a comprehensive and diverse understanding of the relationships between the concepts in various everyday scenarios.

DimonGen is a challenging task because it requires generating reasonable scenarios for a given pair of concepts without any context. This requires a deep understanding of relational and commonsense knowledge about the concepts. Additionally, the target outputs must reflect diverse relationships

[1]Code and data are available at https://github.com/liuchenzhengyi/DimonGen. *Equal contribution. †Corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 4719–4731

between the input concepts. Previous approaches to generating diverse content have used sampling from a designed vocabulary distribution (Holtzman et al., 2020; Meister et al., 2022; Fan et al., 2018) or encoding inputs to various latent variables (Zhao et al., 2017; Cao and Wan, 2020). However, these methods introduce diversity *only* at the generation stage which may not be suitable for the DimonGen task as it relies on the semantic information from the input contexts.

To overcome the challenges, we propose ***MoREE***: ***M**ixture **o**f **R**etrieval-**E**nhanced **E**xperts*, a two-stage method that utilizes external knowledge to generate diverse relationship sentences. In the first stage, MoREE retrieves diverse context sentences related to the given concepts using a mixture of retrievers model based on the Mixture of Experts (MoE) model (Shen et al., 2019). In the second stage, MoREE generates diverse relationship sentences conditioned on the retrieved contexts using a mixture of generators model. An Expectation-Maximization (EM) based matching algorithm is proposed to combine the two stages. By extracting diverse contexts from corporas before generation, MoREE aims to improve the diversity and quality of the generated relationship sentences.

We build a benchmark dataset for DimonGen by adapting the existing CommonGen benchmark (Lin et al., 2020) and conduct both quantitative and qualitative experiments on the dataset. The results indicate that our proposed MoREE model outperforms well-designed baselines in terms of both the quality and diversity of the generated sentences. For example, in the automatic evaluation, our method gains over 2% in the *BLEU-4* score for quality and around 5% in *Self-BLEU-4* for diversity. And in our human evaluation, the annotated score (up to 5) for quality increases from 3.77 to 4.21, and for diverse increases from 3.65 to 3.94. We also conduct detailed ablation studies and case studies to further verify the effectiveness of our proposed method. Overall, the results suggest that MoREE can generate diverse sentences that reflect relationships between concepts from multiple and varied perspectives.

2 DimonGen: Diversified Generative Commonsense Reasoning

We propose a task called *DimonGen* that aims to generate diverse sentences that describe the relationships between a pair of concepts from different perspectives. The task is defined as a diverse constrained text generation task, where the input is a pair of concepts (i.e., $\boldsymbol{x} = \{\boldsymbol{e}_a, \boldsymbol{e}_b\}$) and the output is a set of sentences $\mathcal{Y} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n\}$ that include both concepts and are diverse in terms of their content (an example is illustrated in Fig. 1).

To solve the above task, we propose a two-stage retrieval-enhanced method named MoREE, which consists of a mixture of retriever and generator models (Fig. 2). This method is based on the Mixture of Experts (MoE) model, which will be reviewed in the following section before introducing MoREE.

2.1 Base Model: Mixtrue of Experts for Diverse Text Generation

The Mixture of Experts (MoE) is an ensemble technique that was originally designed to increase the capacity of a model (Jacobs et al., 1991; Jordan and Jacobs, 1994). It consists of several expert models that share the same network architecture, but have different probabilities of being assigned to the same training examples. This means that each expert model is exposed to a different subset of the training data, and the MoE ensemble combines them to achieve optimal performance.

In recent years, MoE has been adapted for text-generation tasks to improve diversity in the generation stage (Shen et al., 2019; Cho et al., 2019). Since the mixture base models are trained on different subsets of the training data, they can learn different aspects of the input, leading to a diverse set of generations during the inference phase. Formally, for each training example $(\boldsymbol{x}, \boldsymbol{y})$ where $\boldsymbol{y} \in \mathcal{Y}$ is a relation description, if there are n expert models with a set of latent variables $\mathcal{Z} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n\}$ as identifiers, the likelihood of the MoE model is formulated as the following marginal likelihood:

$$p(\boldsymbol{y}|\boldsymbol{x};\theta) = \sum_{i=1}^{n} p(\boldsymbol{z}_i|\boldsymbol{x};\theta)p(\boldsymbol{y}|\boldsymbol{z}_i,\boldsymbol{x};\theta), \quad (1)$$

where θ represents the model weights.

To promote diversity among the different expert models, the training examples are split into subsets with distinct elements, and each expert model is trained on one subset. This training process is done through a hard-EM algorithm as follows (Shen et al., 2019; Yu et al., 2022):

- **E-step:** for each training example $(\boldsymbol{x}, \boldsymbol{y})$, select the expert model $z_i \in \mathcal{Z}$ that maximizes

the posterior probability $p(\boldsymbol{z}_i|\boldsymbol{x}, \boldsymbol{y}; \theta)$ using current model weights θ with the equation $\boldsymbol{z}_i = \arg\max_{\boldsymbol{z}\in\mathcal{Z}} p(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{x}; \theta)$.

- **M-step:** update the model weights θ through the gradients $\nabla_\theta \log p(\boldsymbol{y}, \boldsymbol{z}_i|\boldsymbol{x}; \theta)$ of selected expert model $\boldsymbol{z}_i$.

The hard-EM algorithm is performed by iterating these two steps. It should be noted that this algorithm can be easily applied to a batch learning algorithm by updating the model weights for each batch during the M-step. Finally, by assuming a uniform prior of expert models, the loss function could be formulated as

$$\mathcal{L} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}[\min_i -\log p(\boldsymbol{y}|\boldsymbol{z}_i, \boldsymbol{x}; \theta)]. \quad (2)$$

2.2 MoREE: Mixture of Retrieval-Enhanced Experts

The DimonGen task poses a significant challenge as it requires relational commonsense reasoning and the generation of diverse content with the minimal input information. Traditional methods for encouraging diverse text generation focus on introducing diversity in the generation stage through diversified decoding or sampling mechanisms (Meister et al., 2022; Fan et al., 2018; Zhao et al., 2017; Cao and Wan, 2020). However, these methods are not suitable for the DimonGen task due to the limited input information and the need for diversified relational reasoning. Our experiments in Sec. 3.3 show that even with powerful pre-trained language models, these methods struggle to solve this task.

To address this challenge, we propose a diversified retrieval-enhanced method named **M**ixture **o**f **R**etrieval-**E**nhanced **E**xperts (MoREE). Our overall framework is illustrated in Fig. 2 which consists of two stages. In the first stage, we use a mixture of retrievers model to extract several sets of diverse context sentences as auxiliary inputs to help with the generation process. In the second stage, we use a mixture of generators model to generate diverse outputs and propose a matching algorithm to assign the appropriate contexts to the target outputs.

2.2.1 Retrieval Stage

To better understand the relationships between given concepts, we introduce a retrieval stage to gather context sentences from external corpora $\mathcal{C}$. Given an input concept pair $\boldsymbol{x} = \{\boldsymbol{e}_a, \boldsymbol{e}_b\}$, we aim to retrieve several diversified sets of relational contexts $\{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$, where $\mathcal{S}_i = \{\boldsymbol{s}_1^i, \ldots, \boldsymbol{s}_k^i\}$ is a set of context sentences containing $\boldsymbol{x}$.

We train the retriever models on a binary classification task. Given a candidate sentence from external knowledge corpora $\boldsymbol{s}_j \in \mathcal{C}$, we concatenate it with the input $\boldsymbol{x}$ and use it as input:

$$\begin{aligned} \boldsymbol{x}_j^{\text{re}} = \ & \text{[CLS]}\ \boldsymbol{x}\ \text{[SEP]}\ \boldsymbol{s}_j\ \text{[SEP]} \\ & \boldsymbol{x} = \boldsymbol{e}_a\ \text{[SEP]}\ \boldsymbol{e}_b, \end{aligned} \quad (3)$$

where [CLS] and [SEP] are special tokens in pre-trained language models. The model's task is to predict a label y_c from $[0, 1]$, indicating the confidence of the candidate sentence being a true relational context for the input concepts. For each input, we use its target output sentences in the dataset as positive examples and randomly sample the same number of negative examples from its retrieved candidate sentences.

To extract diversified contexts for each input concept pair, we introduce the mixture of experts (MoE) method into the retriever model. Since independently parameterizing each expert may cause an overfitting problem, we follow the weight-sharing schema in Shen et al. (2019) with a unique identifier to solve this issue. To make the MoE models more easily understood by pre-trained language models, for each expert model, we design its unique identifier as latent variables $\boldsymbol{z}_i = z_1^i, \ldots, z_m^i$ which is a randomly sampled prefix token sequence in the model vocabulary. Once an expert is chosen, we could train the model by concatenating the latent variable and input concepts with contexts as the final input:

$$\boldsymbol{x}_{ji}^{\text{re}} = \boldsymbol{z}_i\ \text{[CLS]}\ \boldsymbol{x}\ \text{[SEP]}\ \boldsymbol{s}_j\ \text{[SEP]}\ . \quad (4)$$

We apply the hard-EM algorithm (Shen et al., 2019; Yu et al., 2022) to train our mixture of retrievers model. For each iteration, at E-step, we assign the expert model to each input; at M-step, we update all the expert models with the assigned inputs. With this process, the total training loss turns into an expectation form:

$$\mathcal{L}_c = \mathbb{E}_{(\boldsymbol{x}_j^{\text{re}}, y_c)}[\min_i -\log p(y_c|\boldsymbol{x}_{ji}^{\text{re}}; \theta)]. \quad (5)$$

However, during experiments, we find the binary classification problem has obvious patterns, and simply applying the hard-EM algorithm may lead to a severe overfitting problem (i.e., one expert always predicts one class label). To solve this

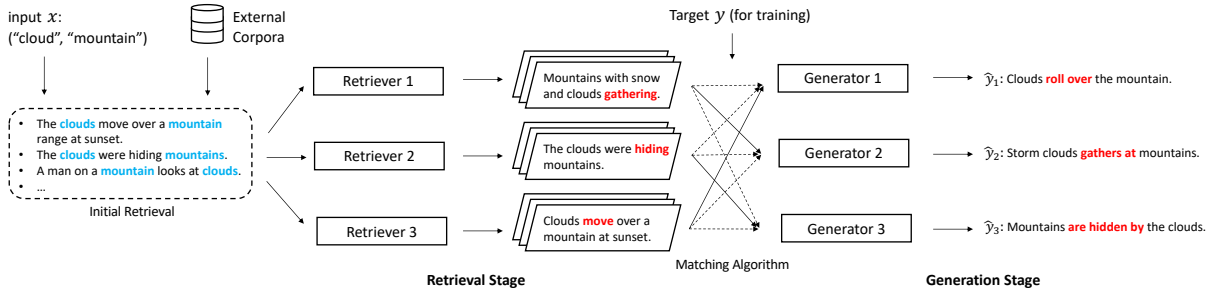

Figure 2: The overall framework of the proposed MoREE method, which includes two stages: 1) retrieval stage with a mixture of retrievers model to extract diversified contexts, 2) generation stage with a mixture of generators model. And a matching algorithm is used to concatenate these two stages.

problem, we propose a regularization term based on Jenson Shannon divergence (Sibson, 1969) to penalize the output probability distribution over different labels among experts. Given the output probability distribution of n experts $\{P_1, \ldots, P_n\}$, the regularization loss is calculated as an average of the Kullback-Leibler (KL) distances between each distribution and the distribution center:

$$\mathcal{L}_r = \frac{1}{n}\sum_{i=1}^{n} D_{\mathrm{KL}}(P_i||\frac{1}{n}\sum_{j} P_j), \qquad (6)$$

where $D_{\mathrm{KL}}(\cdot||\cdot)$ is the KL divergence.

The final loss function for our mixture of retrievers model is a weighted sum of the two:

$$\mathcal{L} = \mathcal{L}_c + \alpha\mathcal{L}_r, \qquad (7)$$

where α is a hyperparameter to balance the two losses.

2.2.2 Generation Stage

At the generation stage, we fine-tune a mixture of generators model to generate qualified and diversified relationship sentences with retrieved sets of contexts. Given an input concept pair $\boldsymbol{x} = \{\boldsymbol{e}_a, \boldsymbol{e}_b\}$ and several diversified sets of context sentences $\{\mathcal{S}_1, \ldots, \mathcal{S}_n\}$ from the retrieval stage, our goal is to generate a set of relationship sentences $\hat{\mathcal{Y}} = \{\hat{\boldsymbol{y}}_1, \ldots, \hat{\boldsymbol{y}}_n\}$.

For each input concept pair $\boldsymbol{x}$ and each set of its context sentences $\mathcal{S}_i = \{\boldsymbol{s}_1^i, \ldots, \boldsymbol{s}_k^i\}$, we concatenate all of their input token sequences with the expert's latent variable $\boldsymbol{z}_i$ to construct the final input as

$$\boldsymbol{x}_i^{\mathrm{gen}} = \boldsymbol{z}_i \text{ [CLS] } \boldsymbol{x} \text{ [SEP] } \boldsymbol{s}_1^i \text{ [SEP] } \ldots \boldsymbol{s}_k^i. \qquad (8)$$

By applying the same method for all sets of retrieved sentences, we can obtain n different context-aware inputs $\{\boldsymbol{x}_1^{\mathrm{gen}}, \ldots, \boldsymbol{x}_n^{\mathrm{gen}}\}$.

However, the retrieved contexts are not present in the original dataset and thus, there is no explicit link between the target outputs and the retrieved contexts. To address this issue, we propose a matching algorithm based on a hard-EM algorithm similar to the one used in the MoE process. For each input, we evaluate its compatibility with each target output by calculating the posterior probability $p(\boldsymbol{y}_j|\boldsymbol{x}_i^{\mathrm{gen}}; \theta)$ using the current generator model's parameters θ. The context-aware input is then assigned to the target output with the highest score:

$$\boldsymbol{y}_i^{\mathrm{target}} = \arg\max_{\boldsymbol{y}_j \in \mathcal{Y}} \; p(\boldsymbol{y}_j|\boldsymbol{x}_i^{\mathrm{gen}}; \theta). \qquad (9)$$

In the training phase, we use Eq. (9) to construct training examples at E-step and then use these examples to fine-tune a mixture of generators model at M-step. In the inference phase, for each input, we feed all the diversified context-aware inputs into the generator model to generate diverse results.

3 Experiments

3.1 Dataset Construction and Analysis

We construct our DimenGen benchmark dataset by combining the CommonGen dataset (Lin et al., 2020), which contains high-quality descriptive sentences for everyday relations between input concepts, and ConceptNet (Speer et al., 2017), a semantic graph with nodes representing concepts and edges indicating the category of the relationship between them. To build our dataset, we first cluster all pairs of input concepts present in the CommonGen dataset and collect their corresponding relational sentences as target relationship sentences. We then verify the informativeness and correctness of the dataset using ConceptNet. Specifically, we ensure that each concept set in every target relationship sentence contains a *path* between the input concept

	train	dev	test
Number	15,263	665	1,181
Unseen ratio (%)	-	91.73	98.31
Avg. ref. number	4.13	3.71	3.38

	3-targets	4-targets	5-targets
ratio (%)	34.76	24.16	41.07

Table 1: The statistics of the DimonGen dataset.

pair on ConceptNet, verifying the existence of a semantic relationship between the concepts described by a chain of category names. This approach helps us establish the semantic relationships between the input concepts in a systematic manner and ensures that the generated sentences contain coherent and meaningful relationships.

To encourage diversity, given a target set of generations for an input concept pair, we first embed each target sentence into latent space with Sentence-BERT (Reimers and Gurevych, 2019) model and calculate the cosine similarity for each pair of them. Next, we filter out the generations that have pair-wise cosine similarity higher than a pre-set threshold $p = 0.75$ in our experiments. For each input concept pair, we limit its target references within $3 \sim 5$. This is because if the number is too small, it is difficult for the model to learn the diversity in the references, while if the number is too large, the models will be trained in a biased manner towards some input concept pairs.

To help evaluate the generalization ability, following CommonGen (Lin et al., 2020), we explicitly control the ratio of unseen concept compositions between input concepts in test examples and target outputs in training examples. Table 1 shows the basic statistics of the dataset. We totally extract 16212 examples in our dataset with 15263, 665, and 1181 split for training, dev, and test. The ratio of unseen concept compositions is 92% and 98% for dev and test respectively. The highly unseen concept compositions make the DimonGen task a difficult problem to solve, which requires the model to be capable of generalized reasoning ability.

For the diversity, the average numbers of target relationship sentences for each example are 4.13, 3.71, and 3.38 for training, dev, and test sets respectively. It is also noted that there are over 41% examples that have 5 target outputs. The high ratio of examples with 5 target references not only contributes to increasing the models' ability to generate diverse outputs but also helps to build comprehensive evaluation metrics.

3.2 Experimental Setup

Baselines. Since we are targeting the DimonGen task with many references for each input, we compare with several strong baseline models with diverse text generation capabilities. Generally, previous works introduce the diversity at the generation stage by either sampling the next word by a probability distribution (Sampling-based methods) or incorporating mixture components in the generator model (MoE-based methods). Different from previous works, our MoREE model introduces diversity by extracting the diverse contexts from the external corpora at the retrieval stage.

- **Sampling-based methods.** Sampling methods create diverse outputs at the inference phase of the generation stage. These methods sample the next token with a designed probability distribution of the vocabulary, rather than simply maximizing the likelihood. We compare with three strong sampling-based methods: Top-k sampling (Fan et al., 2018) truncates the sampling pool by keeping only the top-k candidates for each token in the generation. Top-p sampling (Holtzman et al., 2020) cuts off the next-token sampling pool from a threshold of the probability mass. Typical sampling (Meister et al., 2022) constrains the generated words to expected information content by shifting the truncation set with a conditional entropy of prior content.

- **MoE-based methods.** MoE-based methods introduce diversity at the training phase of the generation stage by using diverse latent variables. We compare with two of them: MoE (Shen et al., 2019) is the vanilla MoE model for diverse text generation we discussed in Sec. 2.1. MoKGE (Yu et al., 2022) incorporates commonsense knowledge from an external graph and uses the MoE model to generate diverse outputs. Compared to our model, MoKGE also extracts information from external knowledge, but it only introduces diversity at the generation stage.

Implementation. In our proposed method, we utilize external corpora from VATEX (Wang et al., 2019), ActivityNet (Krishna et al., 2017), SNLI (Bowman et al., 2015), and MNLI (Williams et al., 2018) for retrieval purposes. These datasets comprise high-quality descriptive sentences and are widely employed in commonsense benchmarking tasks. We retrieve all sentences containing

Method		Quality (top-k) ↑			Pairwise diversity ↓		Corpus diversity ↑	
		BLEU-4	ROUGE-l	S. R.	self-B.-4	self-R.-l	Entropy-4	Distinct-4
Sampling methods	**Top_k sampling**	14.97	40.29	87.75	38.54	61.27	9.50	74.53
	Top_p sampling	15.35	40.17	87.30	33.58	56.57	9.60	78.22
	Typical sampling	15.26	40.42	87.60	35.05	57.99	9.58	77.36
MoE methods	**MoE**	16.70	40.88	87.84	30.86	51.16	9.49	75.87
	MoKGE	16.60	41.34	88.37	29.73	50.02	9.58	79.12
	MoREE (ours)	**19.06**	**43.17**	**91.69**	**24.85**	**46.85**	**9.70**	**83.62**

Table 2: Results of DimonGen task for different methods; evaluation metrics contain three dimensions; "S.R.", "self-B.-4", and "self-R.-l" are abbreviations for "Successful Rate", "self-BLEU-4", and "self-ROUGE-l" respectively (note the lower the pairwise diversity score "↓", the better the performance on diversity).

both input concepts to create a candidate pool. In cases where there are insufficient candidates, we substitute the concepts with the smallest cosine similarity in each sentence, according to their Word2Vec (Mikolov et al., 2013) embeddings. We use pre-trained Roberta models (Liu et al., 2019) as its base model to rank and select the candidates in the retrieval stage. For the generation stage, we use the pre-trained BART model (Lewis et al., 2020) as the base model for all baseline methods and our proposed method for a fair comparison. We require each method to generate $k = 3$ relationship sentences in our experiments because the minimum reference sentences' number in the dataset is 3.

We use Huggingface's Transformers (Wolf et al., 2020) to implement the code and perform a grid search to find the best hyper-parameters for all baseline methods. Our models were trained by one NVIDA RTX A40 GPU card with about 4-5 hours of training on the DimonGen dataset.

Metrics. To evaluate the performance of our proposed DimonGen task, we use three different evaluation metrics: quality, pairwise diversity, and corpus diversity.

- **Quality metrics.** For quality evaluation, we use both N-gram-based metrics such as *BLEU* (Papineni et al., 2002) and *ROUGE* (Lin, 2004), as well as the concept overlapping rate (*Success Rate*) between the input and generated sentences. We make a slight modification for the DimonGen task by first requiring the model to generate a set of top-k candidates, then evaluating the quality between each generated candidate and the target references. The best candidate with the highest score is chosen and its score is used for the quality metrics.

- **Pairwise diversity.** To measure pairwise diversity, we compute the average score of N-gram-based evaluation metrics between all pairs of generations in the generated candidate set. The lower the average score is, the higher the evaluated pairwise diversity will be. These metrics are named *Self-BLEU* and *Self-ROUGE* (Zhu et al., 2018).

- **Corpus diversity.** To evaluate the corpus diversity of the generated text, we use two widely-used metrics: *Distinct-n*(Li et al., 2016) and *Entropy-n*(Zhang et al., 2018). *Distinct-n* is computed by taking the ratio of the number of unique n-grams to the total number of n-grams in the generated sentences. On the other hand, *Entropy-n* calculates the average uncertainty of the n-gram distribution within one generation, providing an estimate of the diversity of the generated text.

3.3 Experimental results

The experimental results in Table 2 show that our proposed MoREE model outperforms all five baseline models in both quality and diversity metrics on the DimonGen task. Specifically, our method achieves a 2% improvement in *BLEU-4* compared to other baseline models in terms of quality, and outperforms the strong baseline MoKGE model by around 5% in *Self-BLEU-4* for pairwise diversity and 4% in *distinct-4* for corpus diversity. These results demonstrate the superior diverse generation capabilities of our proposed method.

Additionally, the results show that MoE-based methods have a significant advantage over sampling-based methods in terms of diversity, with an approximate 5% improvement in *Self-BLEU-4* and 6% in *distinct-4*. Furthermore, retrieving from external corpora improves performance on concept-related evaluation metrics, as shown by the superior *success rate* of the MoKGE model and MoREE model compared to the vanilla MoE model. Our MoREE method specifically achieves a 4% gain in this metric, indicating the effectiveness of the mixture retriever in extracting high-quality contexts to assist diverse generations.

Method	Quality (top-k) ↑			Pairwise diversity ↓		Corpus diversity ↑	
	BLEU-4	ROUGE-l	S. R.	self-B.-4	self-R.-l	Entropy-4	Distinct-4
MoREE	**19.06**	**43.17**	91.69	**24.85**	**46.85**	**9.70**	**83.62**
w/o mixture of retrievers	16.91	41.84	91.04	27.77	50.43	9.54	80.69
w/o regularization term	18.57	42.88	**91.87**	29.40	51.45	9.55	79.31
w/o matching algorithm	16.64	41.78	91.27	28.98	50.96	9.48	77.47

Table 3: Ablation study of our proposed method by taking off each component: 1) mixture of retrievers, 2) regularization term, 3) matching algorithm respectively.

Method	Quality	Diversity	Gra & Flu
DimonGen	**4.70**	**4.25**	**4.67**
Typical	3.65	3.12	4.35
MoKGE	3.77	3.65	**4.63**
MoREE (ours)	**4.21**	**3.94**	4.61

Table 4: Qualitative evaluation results of the DimonGen dataset and generations from three different methods

3.4 Ablation Study

In order to gain a deeper understanding of our proposed two-stage framework, we conduct an ablation study by removing different components of our method and comparing the results. Specifically, we remove the MoE module from the retrieval stage, remove the proposed regularization term for training MoE retrievers, and replace the EM-based matching algorithm for the generation stage with random selection. Table 3 displays the results, revealing the following insights:

- For the retrieval stage, employing a **mixture of retrievers** improves both quality and diversity. When using a single retriever model with MoE generators, the *BLEU-4* score drops from 19.06 to 16.91, and the *Self-BLEU-4* score increases from 24.85 to 27.77. This suggests incorporating diversity into the retrieval stage with a mixture of retrievers can enhance diverse commonsense reasoning capabilities.

- For the retrieval stage, the proposed **regularization term** significantly boosts diversity. Without the regularization term in the loss function during the training process, the *Self-BLEU-4* score increases from 24.85 to 29.40. This demonstrates that our proposed regularization term helps the retriever balance the distribution of different models, which in turn improves the diversity of the retrieved contexts and generations.

- For the generation stage, the proposed **matching algorithm** greatly enhances both quality and diversity. Specifically, our proposed EM-based matching algorithm for matching retrieved contexts to target output gains over 2% on the *BLEU-4* score and 6% on the *distinct-4* score compared to random selection. This indicates that the matching algorithm can effectively assign appropriate contexts to generations, improving the quality and diversity of the generations.

3.5 Human Evaluation

In order to understand the effectiveness of our proposed MoREE method, we conduct the human evaluation by asking three annotators to assign grades (up to 5) of the generated relationship sentences. We randomly sample 100 examples from the test set of the DimonGen dataset and compare our method with the typical sampling and MoKGE methods. Following Yu et al. (2022), we design three evaluation dimensions: quality, diversity, and grammar & fluency (gra & flu).

The human evaluation results in Tab. 4 shows that the DimonGen dataset receives high scores for quality and diversity, indicating that the majority of examples in the dataset are well-written and diverse. Our proposed MoREE method outperforms the two baseline methods in terms of quality and diversity and achieves similar scores for grammar and fluency. This demonstrates that our method is able to effectively capture the complex relationships between concepts in real-world scenarios while also generating a variety of unique and accurate relationship sentences.

3.6 Case study

Table 5 illustrates some generation examples for input concept pairs {"dog", "sheep"} and {"airport", "way"} with different methods, which shows that:

- For the input pair "dog" and "sheep", the generations produced by the baseline methods contain some unreasonable outputs, such as "Sheep and dogs are grazing in a meadow." In contrast, our proposed MoREE method generates more reasonable and diverse outputs, such as "The dog is

Method \ Input	("dog", "sheep")	("airport", "way")
Typical sampling	• A man is walking along a road with a dog and two sheep. • A group of sheep and a dog are grazing on the grass. • A man and a dog are standing in a field with sheep.	• A plane is on its way to an airport. • An airplane is making its way down the runway at an airport. • A motorcade makes its way down the runway at an airport.
MoKGE	• a dog is eating a sheep. • Sheep and dogs are grazing in a meadow. • A dog is walking around a field with sheep.	• a woman makes her way through the airport. • passengers make their way through the airport. • A woman is making her way through an airport.
MoREE (ours)	• The dog is herding sheep with a farmer nearby. • A dog is chasing a flock of sheep. • The dog follows the sheep through the gate.	• passengers at an airport are carrying their luggage to and from the terminal as they make their way to their destinations. • A plane is on its way to the airport. • A plane is making its way down the runway at an airport.
DimonGen (Gold)	• A dog herds a flock of sheep together. • dogs guard the sheep on the mountain pasture. • The dogs are attacking a sheep.	• People make their way off a plane toward the airport. • There is a gray and red plane on the run way at the airport. • US Airways plane moves on a taxi way near its gate at an airport.

Table 5: Generated examples for input concept pairs ("dog", "sheep") and ("airport", "way")

heading sheep with a farmer nearby" and "A dog is chasing a flock of sheep."

- For the input pair "airport" and "way", the baseline methods tend to generate plain and repetitive outputs, such as "Passengers make their way through the airport." In contrast, our proposed MoREE method can accurately capture the relationships between concepts, for example, "A plane is on its way to the airport."

4 Related Work

Generative relational reasoning attempts to generate a coherent sentence involving a pair or a set of concepts/entities (Lin et al., 2020; Huang et al., 2022a,c; Huang and Chang, 2022b). For instance, Lin et al. (2020) introduce *CommonGen*, which aims to generate a coherent sentence that describes an everyday scenario involving a given set of common concepts. Huang et al. (2022a) propose *Open Relation Modeling*, which aims to generate an informative sentence describing relationships between concepts. However, these methods do not consider the diversity of possible relationships that can exist between concepts, leading to a limited understanding of relationships between the concepts.

Incorporating diversity at inference phrase is achieved by sampling methods. Instead of selecting the next token based on maximum likelihood (Freitag and Al-Onaizan, 2017), tokens are sampled from a probability distribution of the vocabulary. For example, Fan et al. (2018) reduces the sampling pool by keeping only the top-k candidates for each token in the generation. Holtzman et al. (2020) limits the next-token sampling pool by a threshold of the probability mass. Meister et al. (2022) restrict the generated words to expected information content by shifting the truncation set with a conditional entropy of prior content. While these methods reduce the training effort for neural models, they are criticized for the low quality of generations (Zhang et al., 2021).

Incorporating diversity at the training phase is achieved through diverse model structures. Specifically, Zhao et al. (2017) propose a conditional variational autoencoder-based framework to embed each input into a latent distribution. Cao and Wan (2020) construct their model based on a conditional generative adversarial network with a diversity loss term. Shen et al. (2019) and Cho et al. (2019) utilize a mixture of experts model to encourage diverse outputs from different expert models. Among previous works, Yu et al. (2022)'s method is most similar to ours. They propose to first extract commonsense knowledge from external knowledge graphs and then use an MoE model to generate diverse outputs. While this work considers incorporating external knowledge to improve generation quality, it falls short in increasing diversity due to the naive retriever shared among all the generators.

5 Conclusion

While previous approaches have used generated sentences to model concept/concept relationships, these methods often rely on a single sentence and can be insufficient or biased in conveying the complexity of these relationships. To address this issue, we propose DimonGen, a task for generating diverse sentences that describe concept relationships in various everyday scenarios. To solve the proposed task, we design a two-stage model called MoREE, which combines a mixture of retriever and generator models. Our experimental results demonstrate the effectiveness of MoREE in generating coherent and diverse sentences to describe concept relationships in everyday scenarios.

Limitations

Our proposed DimonGen task involves generating several diverse sentences to describe the relationships between concepts. However, it does not take into account the number of relationships between different concept pairs. This can lead to problems when applying the model trained on the DimonGen dataset to other unseen concept pairs. For example, some concepts may have a small number of relationships, and asking the model to generate a greater number of diverse relationships may lead to *hallucinations* which can be misleading when using the generative model for educational purposes. We leave this as a future work for the research community.

Additionally, the performance of the MoREE model is heavily dependent on the quality of the external corpora used in the retrieval stage. If the corpora do not contain any relevant information for the input concepts, the MoREE model will perform similarly to a vanilla MoE model. An alternative approach is to retrieve information from the Web (Huang et al., 2022b; Lazaridou et al., 2022).

Last, it should be noted that the base models used in this study were relatively small. Recent studies have demonstrated that large language models possess superior reasoning abilities compared to their smaller counterparts (Wei et al., 2022; Huang and Chang, 2022a). Future work on exploring the diversified generative commonsense reasoning ability of large language models is encouraged.

Acknowledgements

This material is based upon work supported by the National Science Foundation IIS 16-19302 and IIS 16-33755, Zhejiang University ZJU Research 083650, IBM-Illinois Center for Cognitive Computing Systems Research (C3SR) and IBM-Illinois Discovery Accelerator Institute (IIDAI), gift grants from eBay and Microsoft Azure, UIUC OVCR CCIL Planning Grant 434S34, UIUC CSBS Small Grant 434C8U, and UIUC New Frontiers Initiative. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the funding agencies.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Yue Cao and Xiaojun Wan. 2020. DivGAN: Towards diverse paraphrase generation via diversified generative adversarial network. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2411–2421, Online. Association for Computational Linguistics.

Jaemin Cho, Minjoon Seo, and Hannaneh Hajishirzi. 2019. Mixture content selection for diverse sequence generation. 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 3121–3131, Hong Kong, China. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Markus Freitag and Yaser Al-Onaizan. 2017. Beam search strategies for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 56–60, Vancouver. Association for Computational Linguistics.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jie Huang, Kevin Chang, Jinjun Xiong, and Wen-mei Hwu. 2022a. Open relation modeling: Learning to define relations between entities. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 297–308, Dublin, Ireland. Association for Computational Linguistics.

Jie Huang and Kevin Chen-Chuan Chang. 2022a. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*.

Jie Huang and Kevin Chen-Chuan Chang. 2022b. Ver: Learning natural language representations for verbalizing entities and relations. *ArXiv preprint*, abs/2211.11093.

Jie Huang, Hanyin Shao, Kevin Chen-Chuan Chang, Jinjun Xiong, and Wen-mei Hwu. 2022b. Understanding jargon: Combining extraction and generation for definition modeling. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3994–4004, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jie Huang, Kerui Zhu, Kevin Chen-Chuan Chang, Jinjun Xiong, and Wen-mei Hwu. 2022c. DEER: Descriptive knowledge graph for explaining entity relationships. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6686–6698, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. 1991. Adaptive mixtures of local experts. *Neural Computation*, 3:79–87.

Michael I. Jordan and Robert A. Jacobs. 1994. Hierarchical mixtures of experts and the em algorithm. *Neural Computation*, 6:181–214.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 706–715. IEEE Computer Society.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

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

Clara Meister, Tiago Pimentel, Gian Wiher, and Ryan Cotterell. 2022. Locally typical sampling.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Tianxiao Shen, Myle Ott, Michael Auli, and Marc'Aurelio Ranzato. 2019. Mixture models for diverse machine translation: Tricks of the trade. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 5719–5728. PMLR.

Robin Sibson. 1969. Information radius. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 14:149–160.

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

Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. 2019. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 4580–4590. IEEE.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume*

1 (Long Papers), pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wenhao Yu, Chenguang Zhu, Lianhui Qin, Zhihan Zhang, Tong Zhao, and Meng Jiang. 2022. Diversifying content generation for commonsense reasoning with mixture of knowledge graph experts. In *Proceedings of the 2nd Workshop on Deep Learning on Graphs for Natural Language Processing (DLG4NLP 2022)*, pages 1–11, Seattle, Washington. Association for Computational Linguistics.

Hugh Zhang, Daniel Duckworth, Daphne Ippolito, and Arvind Neelakantan. 2021. Trading off diversity and quality in natural language generation. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 25–33, Online. Association for Computational Linguistics.

Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. 2018. Generating informative and diverse conversational responses via adversarial information maximization. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 1815–1825.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–664, Vancouver, Canada. Association for Computational Linguistics.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 1097–1100. ACM.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
In Section 6

☑ A2. Did you discuss any potential risks of your work?
In Section 6

☑ A3. Do the abstract and introduction summarize the paper's main claims?
In Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

In Section 2 and Section 3

☑ B1. Did you cite the creators of artifacts you used?
In Section 1, Section 2, and Section 3

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
In Section 1, Section 3, and the submitted code folder

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
In Section 1, Section 3, and the submitted code folder

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
In Section 1, Section 3, and the submitted code folder

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
In Section 1, Section 3, and the submitted code folder

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 3, and the submitted code folder

C ☑ Did you run computational experiments?

Section 3

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 3

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 3

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 3

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
Section 3

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Section 3

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Section 3, and the submitted code folder

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Section 3, and the submitted code folder

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Section 3, and the submitted code folder

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Section 3