Enhancing Topic-to-Essay Generation with External Commonsense Knowledge

Pengcheng Yang[1,2]*, **Lei Li**[3]*, **Fuli Luo**[2], **Tianyu Liu**[2], **Xu Sun**[1,2]
[1]Deep Learning Lab, Beijing Institute of Big Data Research, Peking University
[2]MOE Key Lab of Computational Linguistics, School of EECS, Peking University
[3]School of Computer Science and Technology, Xidian University
{yang_pc, luofuli, tianyu0421, xusun}@pku.edu.cn, tobiaslee@foxmail.com

Abstract

Automatic topic-to-essay generation is a challenging task since it requires generating novel, diverse, and topic-consistent paragraph-level text with a set of topics as input. Previous work tends to perform essay generation based solely on the given topics while ignoring massive commonsense knowledge. However, this commonsense knowledge provides additional background information, which can help to generate essays that are more novel and diverse. Towards filling this gap, we propose to integrate commonsense from the external knowledge base into the generator through dynamic memory mechanism. Besides, the adversarial training based on a multi-label discriminator is employed to further improve topic-consistency. We also develop a series of automatic evaluation metrics to comprehensively assess the quality of the generated essay. Experiments show that with external commonsense knowledge and adversarial training, the generated essays are more novel, diverse, and topic-consistent than existing methods in terms of both automatic and human evaluation.

为了填补这一空白，我们建议通过动态记忆机制将来自外部知识库的常识整合到生成器中。

此外，采用基于多标签鉴别器的对抗性训练来进一步改进主题一致性。

1 Introduction

Automatic **t**opic-to-**e**ssay **g**eneration (TEG) aims at generating novel, diverse, and topic-consistent paragraph-level text given a set of topics. It not only has plenty of practical applications, e.g., benefiting intelligent education or assisting in keyword-based news writing (Leppänen et al., 2017), but also serves as an ideal testbed for controllable text generation (Wang and Wan, 2018).

Despite its wide applications described above, the progress in the TEG task lags behind other generation tasks such as machine translation (Bahdanau et al., 2014) or text summarization (Rush et al., 2015). Feng et al. (2018) are the first to propose the TEG task and they utilize coverage vector to incorporate topic information for essay generation. However, the model performance is not satisfactory. The generated essays not only lack novelty and diversity, but also suffer from poor topic-consistency. One main reason is that the source information is extremely insufficient compared to the target output on the TEG task. We summarize the comparison of information flow between the TEG task and other generation tasks in Figure 1. In machine translation and text summarization, the source input provides enough semantic information to generate the desired target text. However, the TEG task aims to generate paragraph-level text based solely on several given topics. Extremely insufficient source information is likely to make the generated essays of low quality, both in terms of novelty and topic-consistency.

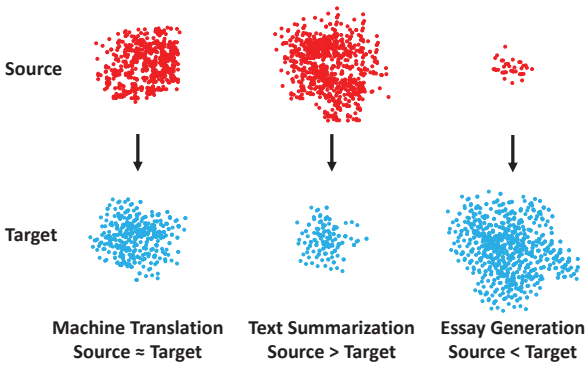

Figure 1: Toy illustration of the information volume on three different text generation tasks, which shows that the source information is extremely insufficient compared to the target output on the TEG task.

In this paper, in order to enrich the source information of the TEG task, we elaborately devise a memory-augmented neural model to incorporate commonsense knowledge effectively. The motivation is that the commonsense from the external knowledge base can provide additional background information, which is of great help to im-

*Equal Contribution.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2002–2012
Florence, Italy, July 28 - August 2, 2019.

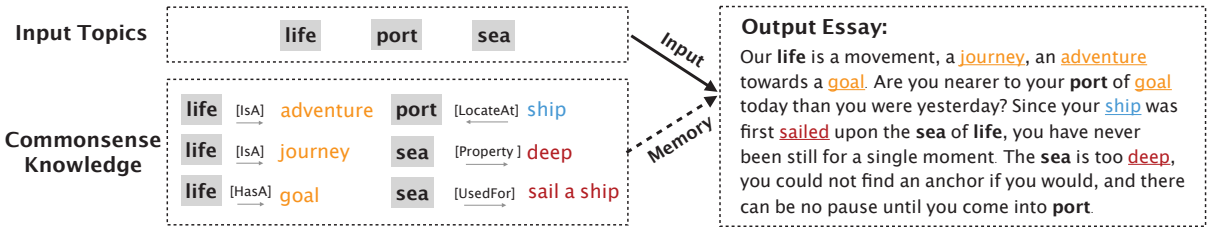

Figure 2: Incorporate commonsense knowledge into topic-to-essay generation via the dynamic memory mechanism. The dashed line indicates that the memory is dynamically updated.

prove the quality of the generated essay. Figure 2 intuitively shows an example. For the given topic "*life*", some closely related concepts (e.g. "*adventure*", "*journey*", "*goal*") are connected as a graph structure in *ConceptNet*[1]. These related concepts are an important part of the skeleton of the essay, which provides additional key information for the generation. Therefore, such external commonsense knowledge can contribute to generating essays that are more novel and diverse. More specifically, this commonsense knowledge is integrated into the generator through the dynamic memory mechanism. In the decoding phase, the model can attend to the most informative memory concepts for each word. At the same time, the memory matrix is dynamically updated to incorporate information of the generated text. This interaction between the memory and the generated text can contribute to the coherent transition of topics. To enhance the topic-consistency, we adopt adversarial training based on a multi-label discriminator. The discriminative signal can comprehensively evaluate the coverage of the output on the given topics, making the generated essays more closely surround the semantics of all input topics.

在解码阶段，该模型可以处理每个单词中信息量最大的记忆概念。

同时，动态更新内存矩阵以合并生成文本的信息

记忆和生成的文本之间的这种交互有助于主题的连贯转换

为了增强主题一致性，我们采用基于多标签判别器的对抗性训练

判别信号可以综合评估输出对给定主题的覆盖率，使生成的文章更紧密地围绕所有输入主题的语义

The main contributions of this paper are summarized as follows:

- We propose a memory-augmented neural model with adversarial training to integrate external commonsense knowledge into topic-to-essay generation.
- We develop a series of automatic evaluation metrics to comprehensively assess the quality of the generated essay.
- Experiments show that our approach can outperform existing methods by a large margin. With the help of commonsense knowledge and adversarial training, the generated essays are more novel, diverse, and topic-consistent.

[1] A large-scale commonsense knowledge base.

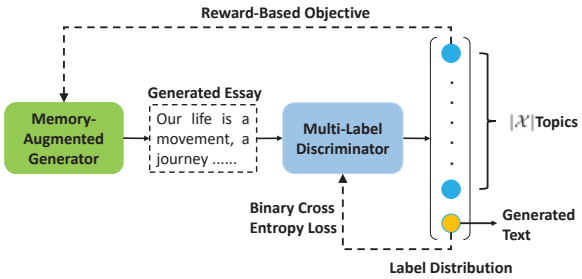

Figure 3: The sketch of our proposed model and adversarial training.

2 Proposed Model

Given a topic sequence $\boldsymbol{x}$ containing m topics, the TEG task aims to generate a topic-consistent essay $\boldsymbol{y}$ containing n words, where n is much larger than m. Figure 3 presents a sketch of our model and training process. The proposed model consists of a memory-augmented generator and a multi-label discriminator. We adopt adversarial training to alternately train the generator and the discriminator.

2.1 Memory-Augmented Generator

The memory-augmented generator G_θ is responsible for generating the desired essay $\boldsymbol{y}$ conditioned on the input topics $\boldsymbol{x}$. Figure 4 illustrates the overview of G_θ, which consists of an encoder and a decoder with the memory mechanism.

Encoder: Here we implement the encoder as an LSTM (Hochreiter and Schmidhuber, 1997) model, which aims to integrate topic information. It reads the input topic sequence $\boldsymbol{x}$ from both directions and computes hidden states for each topic,

$$\overrightarrow{h}_i = \overrightarrow{\text{LSTM}}(\overrightarrow{h}_{i-1}, e(x_i)) \quad (1)$$

$$\overleftarrow{h}_i = \overleftarrow{\text{LSTM}}(\overleftarrow{h}_{i+1}, e(x_i)) \quad (2)$$

where $e(x_i)$ is embedding of x_i. The final hidden representation of the i-th topic is $h_i = [\overrightarrow{h}_i; \overleftarrow{h}_i]$, where semicolon represents vector concatenation.

Decoder: External commonsense knowledge can enrich the source information, which helps

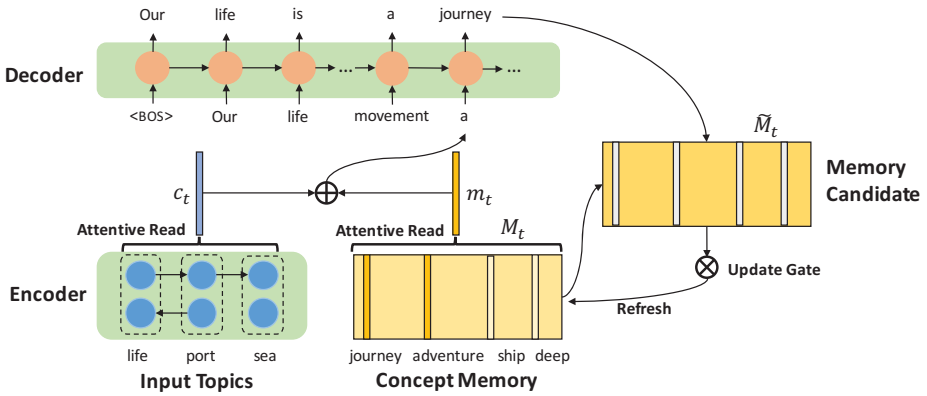

Figure 4: The overview of our memory-augmented generator G_θ. At time-step t, the decoder attends to the concept memory and topic representations to generate a new word. In addition, the memory matrix is dynamically updated via the adaptive gate mechanism.

generate essays that are more novel and diverse. Therefore, we equip the decoder with a memory mechanism to effectively incorporate commonsense knowledge from *ConceptNet*. *ConceptNet* is a semantic network which consists of triples $\mathcal{R} = (\mathtt{h}; \mathtt{r}; \mathtt{t})$ meaning that head concept $\mathtt{h}$ has the relation $\mathtt{r}$ with tail concept $\mathtt{t}$. Since the commonsense knowledge of each topic can be represented by its neighboring concepts in the knowledge base, we use each topic as the query to retrieve k neighboring concepts. The pre-trained embeddings of these concepts are stored as commonsense knowledge in a memory matrix $\mathbf{M}_0 \in \mathbb{R}^{d \times mk}$, where d is the dimension of the embedding vector.[2] In the decoding phase, the generator G_θ refers to the memory matrix for text generation. Specially, the hidden state s_t of the decoder at time-step t is:

$$s_t = \mathrm{LSTM}\big(s_{t-1}, [e(y_{t-1}); c_t; m_t]\big) \quad (3)$$

where $[e(y_{t-1}); c_t; m_t]$ means the concatenation of vectors $e(y_{t-1})$, c_t, and m_t. y_{t-1} is the word generated at time-step $t-1$. c_t is the context vector that is computed by integrating the hidden representations of the input topic sequence,

$$e_{t,i} = f(s_{t-1}, h_i) \quad (4)$$

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{j=1}^{m} \exp(e_{t,j})} \quad (5)$$

$$c_t = \sum_{i=1}^{m} \alpha_{t,i} h_i \quad (6)$$

[2]In practice, the number of columns in $\mathbf{M}_0$ is fixed to K. Supposing there are m input topics, then each topic is assigned $[K/m]$ concepts. For special cases where the concept is insufficient, the pre-trained word2vec embeddings are used as an alternative.

where $f(s_{t-1}, h_i)$ is an aligned model (Bahdanau et al., 2014), which measures the dependency between s_{t-1} and h_i.

m_t in Eq. (3) is the memory vector extracted from $\mathbf{M}_t$, which aims to encode the commonsense knowledge to assist in essay generation. Inspired by Sukhbaatar et al. (2015), we use the attention mechanism to find the rows in $\mathbf{M}_t$ that are most relevant to the output. Formally,

$$v_t = \tanh(\mathbf{W} s_{t-1} + b) \quad (7)$$

$$q_t = \mathrm{softmax}(v_t^{\mathrm{T}} \mathbf{M}_t) \quad (8)$$

$$m_t = \sum_i q_t^i \mathbf{M}_t^i \quad (9)$$

where $\mathbf{W}$ and b are weight parameters. $\mathbf{M}_t^i$ is the i-th column of $\mathbf{M}_t$ and q_t^i is the i-th value of q_t.

Dynamic Memory: As the generation progresses, the topic information that needs to be expressed keeps changing, which requires the memory matrix to be dynamically updated. In addition, the dynamic memory mechanism enables the interaction between the memory and the generated text, which contributes to the coherent transition of topics in the generated essay. Concretely, for each memory entry $\mathbf{M}_t^i$ in $\mathbf{M}_t$, we first compute a candidate update memory $\widetilde{\mathbf{M}}_t^i$,

$$\widetilde{\mathbf{M}}_t^i = \tanh\big(\mathbf{U}_1 \mathbf{M}_t^i + \mathbf{V}_1 e(y_t)\big) \quad (10)$$

where $\mathbf{U}_1$ and $\mathbf{V}_1$ are trainable parameters. Inspired by Highway network (Srivastava et al., 2015), we adopt the adaptive gate mechanism to determine how much the i-th memory entry should be updated,

$$g_t^i = \mathrm{sigmoid}\big(\mathbf{U}_2 \mathbf{M}_t^i + \mathbf{V}_2 e(y_t)\big) \quad (11)$$

Algorithm 1 Adversarial training algorithm.

Require: the memory-augmented generator G_θ; multi-label discriminator D_ϕ; the training corpus $\mathcal{S} = \{(\boldsymbol{x}, \boldsymbol{y})\}$

Initialize G_θ, D_ϕ with random weights θ, ϕ.
Pre-train G_θ using MLE on $\mathcal{S}$
Generate negative samples using G_θ
Pre-train D_ϕ via minimizing Eq. (18)
repeat
  for g-steps **do**
    Generate a sequence $\boldsymbol{y} = (y_1, \ldots, y_n) \sim G_\theta$
    for t in $1 : (n-1)$ **do**
      Compute $r(\boldsymbol{y}_{1:t}, y_{t+1})$ by Eq. (16)
    end for
    Calculate the gradient $\nabla_\theta J(\theta)$ by Eq. (15)
    Update generator parameters
  end for
  for d-steps **do**
    Generate negative examples using G_θ
    Train discriminator D_ϕ via minimizing Eq. (18)
  end for
until Converges

where $\mathbf{U}_2$ and $\mathbf{V}_2$ are learnable parameters. $\mathbf{M}_t^i$ is eventually updated to

$$\mathbf{M}_{t+1}^i = \left(\mathbf{1} - g_t^i\right) \odot \mathbf{M}_t^i + g_t^i \odot \widetilde{\mathbf{M}}_t^i \tag{12}$$

where $\mathbf{1}$ refers to the vector with all elements 1 and $\odot$ denotes pointwise multiplication.

2.2 Multi-Label Discriminator

The discriminator D_ϕ is introduced to evaluate topic-consistency between the input topics and the generated essay, which further improves the text quality. Since the source input contains a variable number of topics, here we implement D_ϕ as a multi-label classifier to distinguish between the real text with several topics and the generated text. In detail, suppose there are a total of $|\mathcal{X}|$ topics, the discriminator produces a sigmoid probability distribution over $(|\mathcal{X}| + 1)$ classes. The score at the i-th ($i \in \{1, \cdots, |\mathcal{X}|\}$) index represents the probability that it belongs to the real text with the i-th topic, and the score at the $(|\mathcal{X}| + 1)$-th index represents the probability that the sample is the generated text. Here we implement the discriminator D_ϕ as a CNN (Kim, 2014) binary classifier.

2.3 Adversarial Training

Inspired by SeqGAN (Yu et al., 2017), here we adopt the adversarial training. We train the memory-augmented generator G_θ via policy gradient method (Williams, 1992). Our generator G_θ can be viewed as an *agent*, whose *state* at time-step t is the current generated words $\boldsymbol{y}_{1:t-1} = (y_1, \cdots, y_{t-1})$ and the *action* is the prediction of the next word y_t. Once the reward $r(\boldsymbol{y}_{1:t-1}, y_t)$ based on both *state* $\boldsymbol{y}_{1:t-1}$ and *action* y_t is observed, the training objective of the generator G_θ is to minimize the negative expected reward,

$$J(\theta) = -\mathbb{E}_{\boldsymbol{y} \sim G_\theta}[r(\boldsymbol{y})] \tag{13}$$

$$= -\sum_{t=1}^{n-1} G_\theta(y_{t+1}|\boldsymbol{y}_{1:t}) \cdot r(\boldsymbol{y}_{1:t}, y_{t+1}) \tag{14}$$

where $G_\theta(y_{t+1}|\boldsymbol{y}_t)$ means the probability that selects the word y_{t+1} based on the previous generated words. Applying the likelihood ratios trick and sampling method, we can build an unbiased estimation for the gradient of $J(\theta)$,

$$\nabla_\theta J(\theta) \approx -\sum_{t=1}^{n-1} \Big\{ \nabla_\theta \log G_\theta(y_{t+1}|\boldsymbol{y}_{1:t}) \cdot r(\boldsymbol{y}_{1:t}, y_{t+1}) \Big\} \tag{15}$$

where y_{t+1} is the sampled word. Since the discriminator can only evaluate a complete sequence, here Monte Carlo Search with roll-out policy G_θ is applied to sample the unknown $n-t$ words. The final reward function is computed as :

$$r(\boldsymbol{y}_{1:t-1}, y_t) = \begin{cases} \frac{1}{N} \sum_{i=1}^{N} D(\boldsymbol{y}_{1:t}^n) & t < n \\ D(\boldsymbol{y}_{1:n}) & t = n \end{cases} \tag{16}$$

where N is the number of searches, $\boldsymbol{y}_{1:t}^n$ is the sampled complete sequence based on the roll-out policy G_θ and *state* $\boldsymbol{y}_{1:t}$, and $D(\boldsymbol{y})$ is defined as:

$$D(\boldsymbol{y}) = \frac{1}{m} \sum_{i=1}^{m} D_\phi(x_i|\boldsymbol{y}) \tag{17}$$

where $D_\phi(x_i|\boldsymbol{y})$ denotes the probability predicted by D_ϕ that the completed sequence $\boldsymbol{y}$ belongs to topic x_i. $D(\boldsymbol{y})$ can be treated as a measure of the coverage of the input topics by the output. A high $D(\boldsymbol{y})$ requires the generated essay to closely surround the semantics of all input topic words.

The discriminator is trained to predict all true topics by minimizing binary cross entropy loss[3],

$$J(\phi) = -\sum_{i=1}^{|\mathcal{X}|+1} \Big\{ x_i \log D_\phi(x_i|\boldsymbol{y}) + (1 - x_i) \log\big(1 - D_\phi(x_i|\boldsymbol{y})\big) \Big\} \tag{18}$$

We alternately train the generator G_θ and the discriminator D_ϕ. An overview of the training process is summarized in Algorithm 1.

[3]When calculating binary cross entropy loss, we convert $\boldsymbol{x}$ into $(|\mathcal{X}| + 1)$-dimensional sparse vector.

3 Experiments

In this section, we introduce the dataset, evaluation metrics, all baselines, and settings in detail.

3.1 Datasets

We conduct experiments on the ZHIHU corpus (Feng et al., 2018). It consists of Chinese essays whose length is between 50 and 100. We select topic words based on the frequency and remove the rare topic words. The total number of labels are set to 100. Sizes of the training set and the test set are 27,000 and 2500. For tuning hyper-parameters, we set aside 10% of training samples as the validation set.

3.2 Settings

We tune hyper-parameters on the validation set. We use the 200-dim pre-trained word embeddings provided by Song et al. (2018). The vocabulary size is 50,000 and batch size is 64. We use a single layer of LSTM with hidden size 512 for both encoder and decoder. We pre-train our model for 80 epochs with the MLE method. The optimizer is Adam (Kingma and Ba, 2014) with 10^{-3} learning rate for pre-training and 10^{-5} for adversarial training. Besides, we make use of the dropout method (Srivastava et al., 2014) to avoid overfitting and clip the gradients (Pascanu et al., 2013) to the maximum norm of 10.

3.3 Baselines

We adopt the following competitive baselines:

SC-LSTM (Wen et al., 2015) uses gating mechanism to control the flow of topic information.

PNN (Wang et al., 2016) applies planning based neural network to generate topic-consistent text.

MTA (Feng et al., 2018) utilizes coverage vectors to integrate topic information. Their work also includes: **TAV** representing topic semantics as the average of all topic embeddings and **TAT** applying attention mechanism to select the relevant topics.

CVAE (Yang et al., 2018b) presents a conditional variational auto-encoder with a hybrid decoder to learn topic via latent variables.

Plan&Write (Yao et al., 2018) proposes a plan-and-write framework with two planning strategies to improve diversity and coherence.

3.4 Evaluation Metrics

In this paper, we adopt two evaluation methods: automatic evaluation and human evaluation.

3.4.1 Automatic Evaluation

The automatic evaluation of TEG remains an open and tricky question since the output is highly flexible. Previous work (Feng et al., 2018) only adopts BLEU (Papineni et al., 2002) score based on n-gram overlap to perform evaluation. However, it is unreasonable to only use BLEU for evaluation because TEG is an extremely flexible task. There are multiple ideal essays for a set of input topics. To remedy this, here we develop a series of evaluation metrics to comprehensively measure the quality of output from various aspects.

Consistency: An ideal essay should closely surround the semantics of all input topics. Therefore, we pre-train a multi-label classifier to evaluate topic-consistency of the output. Given the input topics $\boldsymbol{x}$, we define the topic-consistency of the generated essay $\hat{\boldsymbol{y}}$ as:

$$Consistency(\hat{\boldsymbol{y}}|\boldsymbol{x}) = \varphi(\boldsymbol{x}, \hat{\boldsymbol{x}}) \tag{19}$$

where φ is Jaccard similarity function and $\hat{\boldsymbol{x}}$ is topics predicted by a pre-trained multi-label classifier. Here we adopt the SGM model proposed in Yang et al. (2018a) to implement the pre-trained multi-label classifier.

Novelty: The novelty of the output can be reflected by the difference between it and the training texts. We calculate the novelty of each generated essay $\hat{\boldsymbol{y}}$ as:

$$\begin{aligned} Novelty(\hat{\boldsymbol{y}}|\boldsymbol{x}) =& 1 - max\{\varphi(\hat{\boldsymbol{y}}, \boldsymbol{y_0})| \\ & (\boldsymbol{x_0}, \boldsymbol{y_0}) \in \mathcal{C}_{\boldsymbol{x}}\} \end{aligned} \tag{20}$$

where φ is Jaccard similarity function and $\mathcal{C}_{\boldsymbol{x}}$ is composed of training samples whose corresponding labels are similar to $\boldsymbol{x}$. Formally,

$$\mathcal{C}_{\boldsymbol{x}} = \{(\boldsymbol{x_0}, \boldsymbol{y_0})|\varphi(\boldsymbol{x}, \boldsymbol{x_0}) > \tau\} \tag{21}$$

where τ is the set threshold.

Diversity: We also calculate the proportion of distinct n-grams in the generated essays to evaluate the diversity of the outputs.

In addition, the BLEU scores of different systems are also reported for reference.

3.4.2 Human Evaluation

We also perform human evaluation to more accurately evaluate the quality of the generated essays. Each item contains the input topics and outputs of different models. Then, 200 items are distributed to 3 annotators, who have no knowledge

Methods	BLEU	Consistency	Novelty	Dist-1	Dist-2
SC-LSTM	5.73	1.98	66.51	0.20	0.69
PNN	5.91	11.25	59.52	1.73	6.92
TAV	6.05	16.59	70.32	2.69	14.25
TAT	6.32	9.19	68.77	2.25	12.17
MTA	7.09	25.73	70.68	2.24	11.70
CVAE	7.46	34.84*	71.28	3.72*	17.92*
Plan&Write	8.69*	32.91	72.17*	2.74	14.29
Proposal	**9.72**	**39.42**	**75.71**	**5.19**	**20.49**
Impv-Best	11.85%	13.15%	4.91%	39.52%	14.34%

Table 1: Results of automatic evaluation. Dist-n evaluates the diversity of the output. The best performance is highlighted in bold and "*" indicates the best result achieved by the baselines.

in advance about which model the generated essays come from. Then, they are required to score the generated essay from 1 to 5 in terms of four criteria: novelty, diversity, coherence, and topic-consistency. For novelty, we use the TF-IDF feature to retrieve 10 most similar training samples to provide references for the annotators.

4 Results and Discussion

In this section, we report the experimental results. Besides, further analysis is also provided.

4.1 Experimental Results

The automatic evaluation results are shown in Table 1. Results show that our approach achieves the best performance in all metrics. For instance, the proposed model achieves 11.85% relative improvement over the best baseline on BLEU score. It demonstrates the effectiveness of our approach in improving the quality of the generated essay. More importantly, in terms of novelty, diversity, and topic-consistency, our model can substantially outperform all baselines.

Table 2 presents the human evaluation results, from which we can draw similar conclusions. It is obvious that our approach can outperform the baselines by a large margin, especially in terms of diversity and topic-consistency. For example, the proposed model achieves improvements of 15.33% diversity score and 12.28% consistency score over the best baseline. The main reason for this increase in diversity is that we integrate commonsense knowledge into the generator through the memory mechanism. This external commonsense knowledge provides additional background information, making the generated essays more novel and diverse. In addition, the adversarial training is employed to increase the coverage of the output on the target topics, which further enhances the topic-consistency.

Methods	Consistency	Novelty	Diversity	Coherence
SC-LSTM	1.67	2.04	1.39	1.16
PNN	2.52	1.96	1.95	2.84
MTA	3.17	2.56	2.43	3.28
CVAE	3.42*	2.87*	2.74*	2.63
Plan&Write	3.27	2.81	2.56	3.36*
Proposal	**3.84**	**3.24**	**3.16**	**3.61**
Impv-Best	12.28%	12.89%	15.33%	7.44%
Correlation	0.83	0.66	0.68	0.72

Table 2: Results of human evaluation. The best performance is highlighted in bold and "*" indicates the best result achieved by baselines. We calculate the Pearson correlation to show the inter-annotator agreement.

4.2 Ablation Study

To understand the importance of key components of our approach, here we perform an ablation study by training multiple ablated versions of our model: without adversarial training, without memory mechanism, and without dynamic update. Table 3 and Table 4 present the automatic and human evaluation results of the ablation study, respectively. Results show that all three ablation operations will result in a decrease in model performance. This indicates that both adversarial training and dynamic memory mechanism can contribute to improving the quality of the output. However, an interesting finding is that the adversarial training and memory mechanism focus on improving different aspects of the model.

Memory mechanism We find that the memory mechanism can significantly improve the novelty and diversity. As is shown in Table 3 and Table 4, compared to the removal of the adversarial training, the model exhibits larger degradation in terms

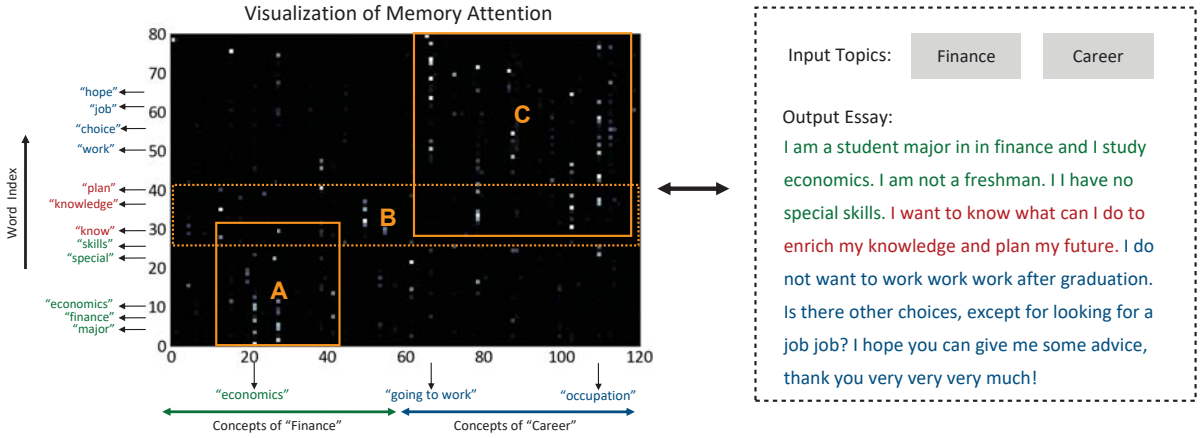

Figure 5: Overview of memory attention during generation. The original Chinese output is translated into English.

Methods	BLEU	Consistency	Novelty	Dist-1	Dist-2
Full Model	9.72	39.42	75.71	5.19	20.49
w/o Adversarial Training	7.74	31.74	74.13	5.22	20.43
w/o Memory	8.40	33.95	71.86	4.16	17.59
w/o Dynamic	8.46	36.18	73.62	4.18	18.49

Table 3: Automatic evaluations of ablation study. "*w/o Dynamic*" means that we use static memory mechanism.

Methods	Consistency	Novelty	Diversity	Coherence
Full model	3.84	3.24	3.16	3.61
w/o Adversarial	3.31	3.07	3.14	3.43
w/o Memory	3.53	2.73	2.77	3.19
w/o Dynamic	3.62	2.91	2.95	3.37

Table 4: Human evaluations of ablation study.

of novelty and diversity when the memory mechanism is removed. This shows that with the help of external commonsense knowledge, the source information can be enriched, leading to the outputs that are more novel and diverse.

Adversarial training Another conclusion is that adversarial training can better benefit the model to enhance the topic-consistency of the generated essay compared to memory mechanism. In detail, Table 4 shows that the consistency score given by humans for ablated versions without adversarial training and memory mechanism decline 0.53 and 0.31, respectively. The reason is that the discriminative signal in training not only evaluates the quality of the generated text, but also models its degree of association with the input topics, thus enhancing the topic-consistency.

4.3 Validity of Memory Module

Here we visualize the attention weights in Eq. (9) to provide a more comprehensive understanding of the memory module. Figure 5 shows an overview of the heatmap of the memory attention weights throughout the process of essay generation.

The attention of coarse-grained topics According to Figure 5, in the early stage of decoding (word index 0 to 30), the generated words focus on the topic "*finance*". In this case, the generator pays more attention to concepts related to "*finance*" (area **A** in the heatmap). As the generation turns more focus on the topic "*career*", some concepts related to "*career*" (area **C** in the heatmap) are assigned larger attention weights. This indicates that our approach can automatically select the most informative concepts based on the topic being focused by the generated text.

The attention of fine-grained words Figure 5 also shows that even focusing on the same topic, our model can finely select the most relevant concepts based on the generated word. For example, when the model generates the word "*finance*" or "*economics*", it pays the most attention to the concept "*economics*". This further demonstrates that the memory module can provide external commonsense knowledge, which does a great favor to the generation of high-quality text.

Coherent transition between topics The dynamic memory can also enhance the coherence of the generated essay. For instance, in the output

Input Topics: *Mother, Childhood*
SC-LSTM: *My hometown is a beautiful city, where the scenery is so beautiful! My hometown has a long history, and the history is so wonderful. Its beauty is beautiful. It is a kind of beauty. It is a kind of beauty.*
PNN: *Childhood is a song. Childhood is a moving song, which carries our laughter. Many, many stories of childhood, childhood memories, just like the stars. Let our childhood be full of happiness. Let us play together, play together.*
MTA: *The mother's love is selfless. It is my mother who gave me life. My mother gave precious life. Let me try to do my best to repay the mother's love. I love my mother. My mother is a great mother!*
CVAE: *My mother is a great. She is very great and she loves me very much. She has given a lot to me. I must love my mother, love my mother in the future.*
Plan&Write: *My mother is very beautiful. She loves me very much. I am very happy with her. I have a good childhood. My happy childhood. I have a good time and let us play together.*
Proposal: *My childhood is a happy family. My mother watches TV at home. I do my homework with my mother. My mother likes to read books, and I am a big fan of books.*

Table 5: Essays generated by different systems. We have translated the original Chinese output into English.

essay in Figure 5, "*I want to know what can I do to enrich my knowledge and plan my future*" is a transition sentence from the topic "*finance*" to the topic "*career*". When generating this sentence, the concepts of both topics (area **B** in the heatmap) receive a certain degree of attention. This illustrates that the dynamic interaction between the memory and the generated text makes the transition between topics more smooth, thus improving the coherence of the output.

4.4 Case Study

Table 5 presents the output of different systems with "*mother*" and "*childhood*" as input topics. As shown in Table 5, the baselines tend to generate low-quality essays. For instance, the output of SC-LSTM and PNN contains massive duplicate phrases. Neither MTA nor CVAE can express information about topic "*childhood*". Although Plan&Write can embody information about both topics, its output is relatively incoherent and less informative. Besides, for the output of these baselines, there exist similar samples in the training set. This indicates that they suffer from poor novelty. Although these baselines strive to incorporate topic information in their unique ways, it is difficult to develop a coherent topic-line based solely on several input topics. This limitation leads to poor coherence and topic-consistency. In contrast, the proposed model succeeds in generating novel high-quality text that closely surrounds the semantics of all input topics. The reason is that our approach can integrate commonsense knowledge into the generator through dynamic memory mechanism. With these additional background information, our model is able to make full expansion to generate the novel and coherent essay. Besides, adversarial training based on the multi-label discriminator further improves the quality of the output and enhances topic-consistency.

5 Related Work

Automatic **t**opic-to-**e**ssay **g**eneration (TEG) aims to compose novel, diverse, and topic-consistent paragraph-level text for several given topics. Feng et al. (2018) are the first to propose the TEG task and they utilize coverage vector to integrate topic information. However, the performance is unsatisfactory, showing that more effective model architecture needs to be explored, which is also the original intention of our work.

A similar topic-to-sequence learning task is Chinese poetry generation. Early work adopts rule and template based methods (Tosa et al., 2008; Yan et al., 2013). When involving in neural networks, both Zhang and Lapata (2014) and Wang et al. (2016) employ recurrent neural network and planning to perform generation. Yan (2016) further propose a new generative model with a polishing schema. To balance linguistic accordance and aesthetic innovation, Zhang et al. (2017) adopt memory network to choose each term from reserved inventories. Yang et al. (2018b) and Li et al. (2018) further utilize conditional variational autoencoder to learn topic information. Yi et al. (2018) simultaneously train two generators via mutual reinforcement learning. However, different from poetry generation presenting obvious structured rules, the TEG task requires generating a long unstructured plain text. Such unstructured target output tends to result in the topic drift problem, bringing severe challenges to the TEG task.

Another similar task is story generation, which aims to generate a story based on the short description of an event. Jain et al. (2017) employ statistical machine translation to explore story generation while Lewis et al. (2018) propose a hierarchical strategy. Xu et al. (2018) utilize reinforcement learning to extract a skeleton of the story to promote the coherence. To improve the diversity and coherence, Yao et al. (2018) present a plan-and-write framework with two planning strategies to fully leverage storyline. However, story generation and the TEG task focus on different goals. The former focuses on logical reasoning and aims to generate a coherent story with plots, while the latter strives to generate the essay with aesthetics based on the input topics. Besides, the source information of the TEG task is more insufficient, putting higher demands on the model.

6 Conclusion

This work presents a memory-augmented neural model with adversarial training for automatic topic-to-essay generation. The proposed model integrates commonsense from the external knowledge base into the generator through a dynamic memory mechanism to enrich the source information. In addition, the adversarial training based on a multi-label discriminator is employed to further enhance topic-consistency. A series of evaluation metrics are also developed to comprehensively assess the quality of the generated essays. Extensive experimental results show that the proposed method can outperform competitive baselines by a large margin. Further analysis demonstrates that with external commonsense knowledge and adversarial training, the generated essays are more novel, diverse, and topic-consistent.

Acknowledgement

We thank the anonymous reviewers for their thoughtful comments. This work was supported in part by National Natural Science Foundation of China (No. 61673028). Xu Sun is the corresponding author of this paper.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *3rd International Conference on Learning Representations*.

Xiaocheng Feng, Ming Liu, Jiahao Liu, Bing Qin, Yibo Sun, and Ting Liu. 2018. Topic-to-essay generation with neural networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 4078–4084.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 2672–2680.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Parag Jain, Priyanka Agrawal, Abhijit Mishra, Mohak Sukhwani, Anirban Laha, and Karthik Sankaranarayanan. 2017. Story generation from sequence of independent short descriptions. *arXiv preprint arXiv:1707.05501*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

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

Leo Leppänen, Myriam Munezero, Mark Granroth-Wilding, and Hannu Toivonen. 2017. Data-driven news generation for automated journalism. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 188–197.

Mike Lewis, Yann Dauphin, and Angela Fan. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 889–898.

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

Juntao Li, Yan Song, Haisong Zhang, Dongmin Chen, Shuming Shi, Dongyan Zhao, and Rui Yan. 2018. Generating classical chinese poems via conditional variational autoencoder and adversarial training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3890–3900.

Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Zhifang Sui, and Xu Sun. 2019. A dual reinforcement learning framework for unsupervised text style transfer. *arXiv preprint arXiv:1905.10060*.

Fuli Luo, Tianyu Liu, Qiaolin Xia, Baobao Chang, and Zhifang Sui. 2018. Incorporating glosses into neural word sense disambiguation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*, pages 2473–2482.

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.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1310–1318.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 175–180.

Robert Speer and Catherine Havasi. 2012. Representing general relational knowledge in conceptnet 5. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, pages 3679–3686.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Highway networks. *arXiv preprint arXiv:1505.00387*.

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. 2015. End-to-end memory networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, pages 2440–2448.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 3104–3112.

Richard S. Sutton, David A. McAllester, Satinder P. Singh, and Yishay Mansour. 1999. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems 12: Annual Conference on Neural Information Processing Systems 1999*, pages 1057–1063.

Naoko Tosa, Hideto Obara, and Michihiko Minoh. 2008. Hitch haiku: An interactive supporting system for composing haiku poem. In *Entertainment Computing - ICEC 2008, 7th International Conference*, volume 5309, pages 209–216.

Ke Wang and Xiaojun Wan. 2018. Sentigan: Generating sentimental texts via mixture adversarial networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 4446–4452.

Zhe Wang, Wei He, Hua Wu, Haiyang Wu, Wei Li, Haifeng Wang, and Enhong Chen. 2016. Chinese poetry generation with planning based neural network. In *26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers*, pages 1051–1060.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Peihao Su, David Vandyke, and Steve J. Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256.

Jingjing Xu, Xuancheng Ren, Yi Zhang, Qi Zeng, Xiaoyan Cai, and Xu Sun. 2018. A skeleton-based model for promoting coherence among sentences in narrative story generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4306–4315.

Rui Yan. 2016. i, poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 2238–2244.

Rui Yan, Han Jiang, Mirella Lapata, Shou-De Lin, Xueqiang Lv, and Xiaoming Li. 2013. i, poet: Automatic chinese poetry composition through a generative summarization framework under constrained optimization. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pages 2197–2203.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018a. SGM: sequence generation model for multi-label classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3915–3926.

Xiaopeng Yang, Xiaowen Lin, Shunda Suo, and Ming Li. 2018b. Generating thematic chinese poetry using conditional variational autoencoders with hybrid decoders. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 4539–4545.

Lili Yao, Nanyun Peng, Weischedel Ralph, Kevin Knight, Dongyan Zhao, and Rui Yan. 2018. Plan-and-write: Towards better automatic storytelling. *arXiv preprint arXiv:1811.05701*.

Xiaoyuan Yi, Maosong Sun, Ruoyu Li, and Wenhao Li. 2018. Automatic poetry generation with mutual reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3143–3153.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 2852–2858.

Jiyuan Zhang, Yang Feng, Dong Wang, Yang Wang, Andrew Abel, Shiyue Zhang, and Andi Zhang. 2017. Flexible and creative chinese poetry generation using neural memory. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1364–1373.

Xingxing Zhang and Mirella Lapata. 2014. Chinese poetry generation with recurrent neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

Yi Zhang, Jingjing Xu, Pengcheng Yang, and Xu Sun. 2018. Learning sentiment memories for sentiment modification without parallel data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1103–1108.